

Evolutionary algorithms and other metaheuristics in water resources: Current status, research challenges and future directions

H.R. Maier^{a,*}, Z. Kapelan^b, J. Kasprzyk^c, J. Kollat^d, L.S. Matott^e, M.C. Cunha^f,
G.C. Dandy^a, M.S. Gibbs^a, E. Keedwell^b, A. Marchi^a, A. Ostfeld^g, D. Savic^b,
D.P. Solomatine^{h,i}, J.A. Vrugt^{j,k,l}, A.C. Zecchin^a, B.S. Minsker^m, E.J. Barbourⁿ,
G. Kuczera^o, F. Pasha^p, A. Castelletti^q, M. Giuliani^q, P.M. Reed^r

^a School of Civil, Environmental and Mining Engineering, The University of Adelaide, Adelaide, SA 5005, Australia

^b College of Engineering, Mathematics and Physical Sciences, University of Exeter, Harrison Building, North Park Road, Exeter EX4 4QF, United Kingdom

^c Civil Environmental and Architectural Engineering Department, University of Colorado Boulder, UCB 428, Boulder, CO 80309, USA

^d DecisionVis, 315 S. Allen St., Ste 321, State College, PA 16801, USA

^e Center for Computational Research, University at Buffalo, Buffalo, NY 14203, USA

^f Department of Civil Engineering, University of Coimbra, Rua Luís Reis Santos – Pólo II, 3030-788 Coimbra, Portugal

^g Civil and Environmental Engineering, Technion – Israel Institute of Technology, Haifa 32000, Israel

^h UNESCO-IHE Institute for Water Education, Delft, The Netherlands

ⁱ Water Resources Section, Delft University of Technology, The Netherlands

^j Department of Civil and Environmental Engineering, University of California Irvine, Irvine, CA, USA

^k Department of Earth System Science, University of California Irvine, Irvine, CA, USA

^l Institute for Biodiversity and Ecosystem Dynamics, University of Amsterdam, Amsterdam, The Netherlands

^m Department of Civil and Environmental Engineering, 205 N. Mathews Ave., MC-250, Urbana, IL 61801, USA

ⁿ Integrated Catchment and Management Centre, Fenner School of Environment and Society, Australian National University, Canberra, ACT 0200, Australia

^o School of Engineering, University of Newcastle, Callaghan, NSW 2308, Australia

^p Department of Civil and Geomatics Engineering, California State University, Fresno, CA 93740, USA

^q Department of Electronics, Information, and Bioengineering, Politecnico di Milano, Piazza L. da Vinci, 32, 20133 Milan, Italy

^r Department of Civil and Environmental Engineering, Cornell University, Ithaca, NY, USA

Received 28 March 2014 Received in revised form 12 September

2014 Accepted 12 September 2014 Available online

* Corresponding author.

E-mail address: holger.maier@adelaide.edu.au (H.R. Maier).

1. Introduction

1.1. Background

Environmental change, economic and social pressures, and limited resources motivate systems analysis techniques that can help planners determine new management strategies, develop better designs and operational regimes, improve and calibrate simulation models, and resolve conflicts between divergent stakeholders. Metaheuristics are emerging as popular tools to facilitate these tasks, and in the field of water resources, they have been used extensively for a variety of purposes (e.g. model calibration, the planning, design and operation of water resources systems etc.) in many different application areas over the last few decades (Nicklow et al., 2010). Since metaheuristics were first applied in the water resources field (Dougherty and Marryott, 1991; McKinney and Lin, 1994; Ritzel et al., 1994; Gupta et al., 1998), their popularity has increased dramatically, probably facilitated by the simultaneous increase of available computational power (Washington et al., 2009), to the point where they are widely used (Nicklow et al., 2010), even by actual water planning utilities (Basdekas, 2014).

Zufferey (2012) defines a metaheuristic “as an iterative generation process which guides a subordinate heuristic by combining intelligently different concepts for exploring and exploiting the search space”, as part of which “learning strategies are used to structure information in order to find efficiently near-optimal solutions.” Unlike more “traditional” approaches, which use mathematical programming to specify the optimal value of one or more objective functions, metaheuristics incorporate elements of structured randomness for search and follow empirical guidelines, often motivated by observations of natural phenomena (Collette and Siarry, 2003).

Metaheuristics can be divided into two groups, including population-based algorithms (e.g. genetic algorithms, evolutionary strategies, particle swarm optimization, ant colony optimization, etc.) and single point-based methods (e.g. simulated annealing, tabu search, simple (1+1) evolutionary strategies, trajectory or local search methods, etc.). Evolutionary algorithms (EAs) are the most well-established class of metaheuristics for solving water resources problems and are inspired by various mechanisms of biological evolution (e.g. reproduction, mutation, crossover, selection, etc.) (Nicklow et al., 2010). Consequently, the focus of the remainder of this paper is on EAs, although many of the concepts discussed also

broadly apply to other metaheuristics. The paper also provides general guidelines and future research directions for the broader class of systems analysis approaches that take any sort of optimisation into account.

When using EAs, the steps in the optimisation process generally include (Fig. 1):

1. Problem formulation (i.e. selection and definition of decision variables, objectives, and constraints).
2. Selection of decision variable values.
3. Evaluation of objectives and constraints for the selected decision variable values, which is generally done using one or more simulation models.
4. Selection of an updated set of decision variable values based on feedback received from the evaluation process using some search methodology.
5. Repetition of points 3 and 4 until the selected stopping criterion has been satisfied.
6. Passing the optimal solutions into an appropriate decision-making process.

As outlined below, compared with more “traditional” optimisation methods, EAs have a number of advantages, which are most likely responsible for their widespread adoption for water resources problems.

1. The basic analogies that inform their optimisation strategies are conceptually easy to understand.
2. As simulation models are generally used to calculate objective function values and check constraints, it is easy to add optimisation to existing simulation approaches. This gives rise to the potential for greater confidence in the results by end users, as the outcomes of the optimisation process are based on the results of simulation tools that are already used for the purposes of decision-making.
3. EAs are capable of solving problems with difficult mathematical properties (Reed et al., 2013). This is because the ability to link with simulation models reduces the need for problem simplification, which is required for many traditional optimisation algorithms that are unable to deal with nonlinearities (e.g. exact finitely terminating algorithms, like linear and nonlinear programming) or discontinuities (e.g. iterative/convergent algorithms, such as first or second order gradient methods). For example, in linear programming applications, there is no ability

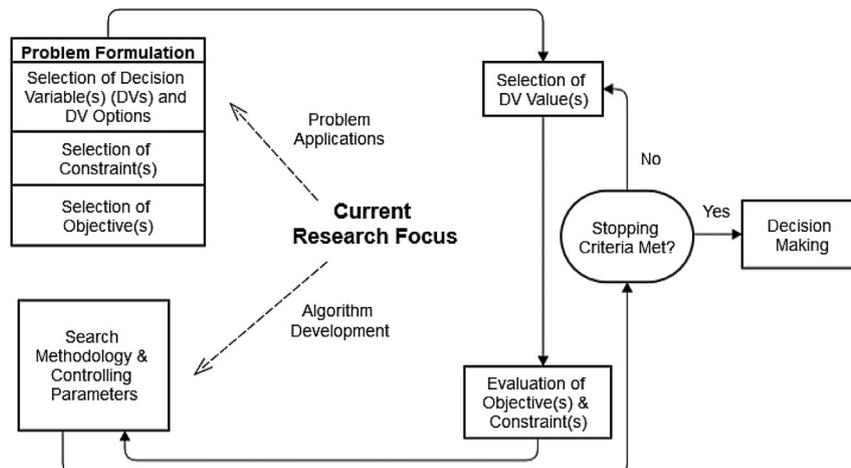


Fig. 1. Steps in EA optimisation process, highlighting areas of current research focus. The square shapes represent the steps in the EA process and the oval shape represents a decision point.

to account for nonlinearities, such as “if-then” style rules. Consequently, the philosophy underpinning EAs is that it is generally better to find near globally optimal solutions to the actual problem, rather than globally optimal solutions to a simplified problem, especially when the simplified problem misses key socially relevant properties (Rittel and Webber, 1973).

4. The linking with simulation models facilitates the straightforward treatment of parallel computing.
5. EAs have the ability to perform both exploration (i.e. global search) and exploitation (i.e. local search) of the fitness function, increasing the chances of finding near-optimal solutions to complex problems (Nanda and Panda, 2014).
6. The algorithms themselves are readily adaptable to a wide variety of application contexts (Back et al., 1997; Goldberg, 1989; Nicklow et al., 2010).

1.2. Purpose and organisation of this position paper

This position paper aims to contribute to the literature that reviews EA algorithms (e.g. Coello et al., 2007; Deb, 2001) and EA use in water resources (e.g. Nicklow et al., 2010), and performs diagnostic assessments on water-related problems (e.g. Reed et al., 2013). However, the primary purpose of this paper is to map out the most important research challenges and future directions in applying EAs to the complex, real-world water resource applications that are most in need of these methods. While a brief review of current progress in relevant areas is provided, it should be noted that this is not meant to be a comprehensive review paper.

The research challenges identified in this paper are motivated by the fact that over the last 20–25 years, much of the research in the field of EAs in water resources has focused on the application of different types of algorithms to different problem types. In the majority of these studies, the aim was either to (Fig. 1):

1. Develop and test the performance of different types of algorithms (e.g. an algorithm inspired by a different natural phenomenon (e.g. genetics, the foraging behaviour of ants in search for food, the behaviour of bees and birds etc.)), variants of existing algorithms, or hybrid algorithms; or
2. Test if a particular algorithm or variant can be used successfully to solve different types of water resources optimisation problems (e.g. model calibration, water distribution system design, groundwater remediation, environmental flow allocation, reservoir operation etc.) and/or different instances of these problem types (e.g. using different models, in different geographical locations etc.).

In relation to the first point, even though studies have shown that certain algorithms perform better than others for selected case studies using particular assessment criteria, our understanding of *why* this is the case is rather limited. Consequently, in order to progress research in this area, there is a need to develop an understanding of the relationship between the *fundamental properties of the problem* (case study) being solved, the *searching behaviour of the optimisation algorithm*, and *algorithm performance*. This will enable us to develop a better understanding of why algorithms exhibiting particular searching behaviours perform better or worse for problems with certain characteristics, rather than simply knowing the relative performance of particular algorithms on specific case studies. Such insight would transcend specific algorithms or case studies to provide understanding that is more generally applicable and open the door to developing guidelines with regard to choosing the most appropriate algorithm(s) for a particular problem. The research challenges that need to be addressed in order to develop an

improved understanding of algorithm performance are summarized in Table 1 and discussed in detail in Section 2.

In relation to the second point, while past research has shown that EAs can be applied successfully to a range of problems in a variety of settings, many of the case studies considered have been academic (Nicklow et al., 2010) and lack a connection to complexities that emerge in potentially contentious and severely uncertain real world decisions. Now that the potential of EAs has been demonstrated, there is a need to tackle a variety of research challenges so that these algorithms may be applied more broadly in real decision contexts, as outlined in Table 2 and discussed in detail in Section 3. This requires an understanding of their mathematical assumptions and their fidelity for addressing changing problem properties. The use of EAs needs to be more broadly viewed through inter-dependent effects of the full suite of choices made in the problem solving process (see Fig. 1 and Table 2). In addition, a major challenge in the existing EA applications literature is that the different steps in the optimisation process (Fig. 1) are not explicitly considered by users and consequently represent poorly understood implicit assumptions. As poorly understood implicit assumptions, these choices can significantly degrade the value of EAs, or even worse, negatively impact the systems focus by ignoring key challenges (see related discussion in Nicklow et al., 2010).

As mentioned above, the current status, research challenges and future directions associated with improving our understanding of algorithm performance (Table 1) are discussed in Section 2, while the current status, research challenges and future directions associated with applying EAs to real-world problems (Table 2) are discussed in Section 3. The appendix provides a glossary for clarifying the terminology adopted in the paper. As this position paper is primarily focused on highlighting ongoing challenges in different areas, it has been organized in a manner that allows for the different topical subsections to be read independently of each other. This structure means that the paper need not be read in a conventional “start-to-finish” linear fashion. Instead, readers can selectively look over different sections based on their own personal interest and quickly navigate to topical sections of interest with the aid of Tables 1 and 2.

2. Understanding of algorithm performance

2.1. Overview

The development of an *understanding* of optimisation algorithm performance is essential to obtaining better insight into *why* certain algorithms perform better on certain types of problems. The development of this understanding requires consideration of the following issues (Fig. 2).

2.1.1. Knowledge of the fundamental characteristics of the problem being optimised at the level at which optimisation algorithms operate

These characteristics are represented by the *fitness landscape*, firstly introduced by Wright (1932), which describes the search

Table 1

Research challenges associated with improving our understanding of algorithm performance.

Section 2.1	Overview of major topics for improving our understanding of algorithm performance.
Section 2.2	Can we develop knowledge of the fundamental characteristics of the problem being optimised at the level at which optimisation algorithms operate?
Section 2.3	Can we develop knowledge of the underlying searching behaviour of different search methodologies?
Section 2.4	How can we rigorously measure and improve the performance of a selected search methodology?

Table 2
Research challenges associated with applying EAs to real-world problems.

Section 3.1	Overview of major topics for applying EAs to real-world problems.
Section 3.2	How do we best change the formulation of optimisation problems to cater to real-world problems?
Section 3.3	What can be done to reduce the size of the search space for real-world problems?
Section 3.4	How can computational efficiency be increased for real-world problems?
Section 3.5	Which searching mechanisms are best for solving real-world problems?
Section 3.6	What termination/convergence criteria are most appropriate for real-world problems?
Section 3.7	What is the best way to convey the results of the optimisation of real-world problems to decision makers and what is the role of optimisation in the decision-making process ?
Section 3.8	What is the best way to take account of uncertainty in the optimisation of realistic systems?
Section 3.9	What is the best way to implement optimisation algorithms for realistic systems?

space of an optimisation problem as a multidimensional landscape defined by the possible solutions, through which the optimisation algorithm moves, mapped to the corresponding fitness value (Smith et al., 2002). As such, the fitness landscape is not only dependent upon the problem to be solved, but also on the choice of algorithm and its parameter values. It should be noted that the fitness landscape is different from the fitness function, which is a more traditional way to consider optimisation problems, where variables of similar values are mapped next to each other, independently from the algorithm adopted. It is the properties of the fitness landscape that determine how easy or difficult it is to solve a particular optimisation problem. For example, the overall structure of the landscape, such as a “big bowl” shape, can guide the algorithm towards the global optimum, while a surface that is rough with many local optima may present difficulties. It is likely that fitness landscapes with vastly different properties can be obtained for the same types of problems (e.g. calibration, water distribution system design etc.), depending on the way problems are formulated, the way constraints are handled, and the simulation model and optimisation algorithm used, and that different types of problems can have fitness landscapes with similar properties. Consequently, it might not be possible to say that particular algorithms are applicable to particular problem types. Instead, the applicability of different algorithms needs to be related to the properties of the fitness landscape of a particular problem. Current progress and remaining challenges surrounding this issue are discussed in Section 2.2.

2.1.2. Knowledge of the underlying searching behaviour of different search methodologies

At a fundamental level, a balance has to be struck in the searching behaviour of algorithms between exploitation/intensification (i.e. convergence to the nearest optimum) and exploration/diversification (i.e. covering different areas of the solutions space in order to find the globally optimal solution). This balance is affected by the type of algorithm used, as well as the values of the parameters that control searching behaviour, such as the selection operator and probability of cross-over and mutation in genetic algorithms. However, at this fundamental level, we have very little understanding of whether the searching behaviour of different algorithms is completely different or whether the same or similar searching behaviour can be achieved by different algorithms if particular parameter combinations are used. In order to claim that an algorithm is effective, we must be able to characterize the **fundamental searching behaviour** of different algorithms with

different parameterisations. Current progress and remaining challenges surrounding this issue are discussed in Section 2.3.

2.1.3. Knowledge of the performance of the selected search methodology

In order to be able to compare the performance of different search methodologies in a rigorous and unbiased fashion, appropriate **performance assessment metrics** need to be used. Current progress and remaining challenges surrounding this issue are discussed in Section 2.4.

2.2. Characterisation of fitness landscape properties

An implication of the No Free Lunch Theorem (Wolpert and Macready, 1997) is that an optimisation algorithm that will produce the best results for all problems does not exist. Intuitively, this is easy to understand if we consider that a local gradient based approach will be most efficient on a smooth, convex, and deterministic objective function, but will be outperformed by global methods on multi-modal problems. There are some aspects of a problem that can be used to guide the selection of an algorithm (e.g. problems with discrete/real valued decision spaces, or the dimensionality of the decision space), however, as applications increase in their complexity (e.g. the use of highly nonlinear simulation models or increasing numbers of risk-based objectives), it can be difficult *a priori* to understand the shape and properties of the objective-space surfaces, and the resultant best suited algorithm. In fact, when averaged across all possible problems, any two optimisation algorithms are expected to be equally effective. Yet, numerous comparisons of optimisation algorithms in the water resources field (as well as more broadly) have demonstrated that for certain subsets of problems, a particular implementation of one algorithm can be found to outperform some others (Kollat and Reed, 2006; Reed et al., 2013) (motivating recent efforts to create auto-adaptive multi-algorithms (Hadka and Reed, 2013; Vrugt and Robinson, 2007; Vrugt et al., 2009)). As a certain algorithm can be found to perform better on a certain problem compared with others, it might be expected that it is the characteristics of the problem being solved that determine when an algorithm may perform better than others. However, formally distinguishing the statistical behaviour of EAs on a given topology is an uncertain process itself that requires rigorous statistical experiments (see Section 2.3 and Back et al., 1997; Bayer and Finkel, 2004; Goh and Tan, 2009; Goldberg, 2002; Hadka and Reed, 2012; Matott et al., 2012). As such, the most common motivation for assessing the problem characteristics through fitness landscape analysis is to gain a better understanding of optimisation algorithm performance on a given set of problems (Pitzer and Affenzeller, 2012).

Jones (1995) and Stadler (2002) provide a formal definition of the fitness landscape, where the landscape is defined as a directed graph, where the nodes correspond to solutions, and the edges (connections) to search operator manoeuvres. Following this, every EA operator induces its own landscape of solutions that can be generated by one application of that operator. Based on this definition, the fitness landscape is not only dependent upon the problem to be solved, but also the choice of algorithm through its parameterised search operators. In some cases, the landscape defined by an operator is strongly related (or even equivalent) to the topological distance between solutions in the decision space (Moraglio and Poli, 2004), for which cases it may be sufficient to study the fitness landscape as defined by the topology of the fitness function (Pitzer and Affenzeller, 2012). Despite the intimate link between a problem's fitness function and the induced algorithm behaviour (as indicated by the connecting arrow in Fig. 2), this section discusses issues associated with the characterisation of the

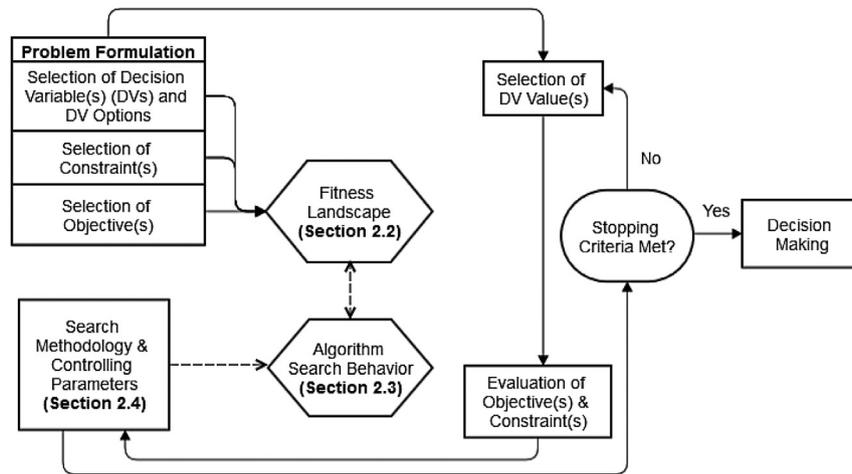


Fig. 2. Research challenges associated with obtaining a better understanding of algorithm performance. The square shapes represent the steps in the EA process, the oval shape represents a decision point and the hexagonal shapes represent problem and algorithm characteristics resulting from choices made at different steps of the EA process.

landscape properties (as they depend on the topology defined by algorithm operators), and Section 2.3 focusses on the temporal characterisation of an algorithm's run-time search behaviour.

2.2.1. Current status

In the field of optimisation problem understanding, fitness landscape analysis is by far the most extensively researched topic, and often forms the basis for research into problem difficulty assignment, algorithm selection and performance prediction (McClymont, 2013). A number of statistics have been developed to measure different characteristics of the fitness landscape, typically based on a sample of solutions and the corresponding fitness function value. A summary of the major characteristics of fitness landscapes that have been identified, and metrics to quantify them, is provided in Table 3. For more information on these characteristics, the reader is directed to two recent reviews on the topic of fitness landscape analysis, including Pitzer and Affenzeller (2012) and Malan and Engelbrecht (2013).

While there are numerous fitness landscape characteristics and corresponding metrics, the statistical reliability of these metrics is frequently called into question. One of the main reasons that these measures have been found to be unreliable is that no rigorous definition of the concept of “difficulty” is available in

the framework of evolutionary optimisation (Kallel, 1998). In this context, difficulty is determined by assessing the likelihood and speed with which an optimisation algorithm can locate a known global optimum. Given this definition, it is clear that the algorithm itself contributes to the “difficulty” of the problem. A number of papers (Culberson and Lichtner, 1996; De Jong et al., 1995; Guo and Hsu, 2003) stress the futility of speaking of problem difficulty without considering the algorithm, and all the parameters of the algorithm (Kallel, 1998). For example, on a given problem, a GA coupled with a local search may find the optimum every time, regardless of the initial population, while the same GA without the local search operator may find the problem very difficult. Studies that attempt to demonstrate the unreliability of fitness function statistics (including Kallel et al., 1999; Kinnear Jr, 1994; Naudts and Kallel, 2000; Quick et al., 1998; Reeves and Wright, 1995; Rochet et al., 1998) have generally drawn conclusions about the convergence of the algorithm, without consideration of the calibration of the algorithm, or the type of algorithm.

2.2.2. Research challenges and future directions

There are a number of outstanding research challenges in the field of fitness landscape analysis, as outlined below.

Table 3

Fitness landscape characteristics, how they influence problem difficulty, and an example metric to quantify the characteristic.

Characteristic	Link to problem difficulty	Example metric
Modality or number of local optima Global structure or ruggedness	The more local optima the more difficult it is to find the best solutions. Overall structure in the landscape can guide the algorithm towards the global optimum.	Iterated Local Search (Merz, 2004) Correlation Measures (Weinberger, 1990)
Neutrality	Neutral areas in the landscape, that are flat and of the same function value, can trap algorithms and limit the ability to find better solutions.	Genetic Distance Operator (Katada and Ohkura, 2006)
Degree of variable interdependency	Referred to as epistasis, where an epistatic function cannot be decomposed into separate functions consisting of single decision variables, and as such combinations of decision variables must be considered.	Epistasis Variance (Davidor, 1991)
Saliency	If one decision variable dominates the fitness function, this variable must converge to a good value before the other decision variables have an influence, which may have converged to poor values in the meantime.	Saliency Measure (Gibbs et al., 2011)
Deception	A deceptive function has the global optimum located away from good local optima, making it more difficult to locate (e.g. needle in a haystack problems).	GA-Deception (Deb and Goldberg, 1994)
Evolvability	The ability of an algorithm to produce better solutions at the next iteration, depending on the problem to be solved.	Fitness Evolvability Portraits (Smith et al., 2002)

2.2.2.1. Link between fitness landscape analysis and algorithm implementation. Ultimately, the aim of fitness landscape analysis is to improve the optimisation process, resulting in the identification of the best possible solution(s). While it is logical that a link between the characteristics of a fitness function and the best algorithm to find optimal solutions should exist, such links have not yet been identified in a way that provides general information to the optimisation process. An example of linking the results from fitness landscape measures to optimisation algorithm parameters is provided by [Gibbs et al. \(2011\)](#). The utilisation of the results of fitness landscape analysis for the identification of the best, or even a suitable, algorithm, still remains a significant research challenge, and is discussed in more detail from the perspective of algorithm search behaviour in Section [2.3.2](#).

2.2.2.2. True fitness landscape, not fitness function, analysis. The mapping of the fitness landscape based on the operators of the optimisation algorithm is a challenge. As a result, most fitness landscape analysis studies have focussed only on single objective problems, and have adopted the decision space, or fitness function, topological distance to be a suitable surrogate. Nonetheless, in order to provide insight into how to select a suitable EA (with different combinations of operations making up the algorithm), it is likely that analysis of the fitness landscape, with topology defined by the operators, will provide the most valuable information. In this way, changes in the landscape after adopting different search operators or different parameter values for an operator, can be assessed (e.g. which operator results in a smoother landscape or one with fewer local optima). How to analyse the true fitness landscape for non-trivial problems using a generalised methodology is an open research question.

2.2.2.3. Is fitness landscape analysis even worthwhile?. An important question to answer is whether fitness landscape analysis is worthwhile, or whether the function evaluations and processing time required for fitness landscape analysis are better spent on actually solving the problem. This is especially the case for water resources problems, where often the fitness function is based on the output of simulation models that have long computation times and a high dimensional search space that requires many evaluations in order to be sampled sufficiently. How to best undertake this sampling is also unclear.

While fitness landscape analysis is usually much more resource intensive than just solving a given problem, the resulting insights can be valuable for increasing problem understanding ([Pitzer and Affenzeller, 2012](#)). As such, fitness landscape analysis may not be the best use of resources in applied studies where the sole aim is to solve a single problem instance, typically to assist in decision making. However, this approach is likely to be highly beneficial in more research focused studies, where a deeper understanding of the link between algorithm performance and problem characteristics, on a subset of problem classes, is of interest ([Pitzer and Affenzeller, 2012](#)).

At least some evaluation of the fitness landscape is required to lead to a deeper understanding of why a particular instance of one optimisation algorithm located better solutions than another, and to move beyond simple algorithm comparative studies with limited potential to be generalised. As [Pitzer and Affenzeller \(2012\)](#) concluded: “*there might not be a free lunch, but at least, there might be some free appetizer that can be reaped more easily using fitness landscape analysis techniques (Droste et al., 1999)*”. Moreover, a generalized understanding of fitness landscapes should also be linked to decision making processes. While some classes of problems may be easier to solve, they may fail to sufficiently encompass problem traits critical to the actual decision-making process ([Brill](#)

[et al., 1990](#); [Haines and Hall, 1977](#); [Liebman, 1976](#); [Reed et al., 2013](#)). Formally, understanding the fitness landscape impacts of the formulation changes required for improving decision relevance can highlight needed algorithm innovations.

2.3. Characterisation of algorithm searching behaviour

The majority of previous research on the comparative analysis of EAs has typically focused exclusively on overall end-of-run performance statistics based on solution quality and computational efficiency ([Nicklow et al., 2010](#)). Despite the governing importance of these statistics, as outlined in Section [2.1](#), this framework of comparison provides little insight into what run-time searching characteristics result in good performance, or on the dynamics of how algorithms utilise their iterative solution generation processes to navigate their way through the decision space in order to: locate and explore feasible regions; exploit high quality information to converge to multiple near-optimal (or Pareto-optimal) regions; and avoid premature convergence. In the aforementioned literature, algorithm operators (or indeed algorithms themselves) are often qualitatively described as exploration or exploitation encouraging ([Colormi et al., 1996](#)), implying a certain kind of impact on an algorithm's searching behaviour. Given these intuitively clear qualitative behavioural descriptors, the basis of a behavioural analysis framework is to provide quantitative measures that capture the important features of an algorithm's run-time search in order to gain insight into the actual influence of an algorithm's search methodology, and its controlling parameters, on its search behaviour. Given the potential for a rigorous behavioural analysis framework to provide deeper understanding of operator influences, and a more informed approach to the application and development of algorithms, continued research to develop and utilise such a framework is imperative. As mentioned in Section [2.2](#), there is a close dependency between an algorithm's manifest behaviour and a problem's fitness landscape characteristics, which is discussed in more detail in Section [2.3.2](#).

2.3.1. Current status

Behavioural measures are loosely defined as metrics that can be computed during an algorithm's run-time, and are based on properties of the current, and historic, set of an algorithm's population of solutions. Such measures typically feature in analytic convergence studies of algorithms (e.g. [Rudolph, 1994](#); [Zaharie, 2002](#)), however, the focus here is on the usage of such measures for computational analysis. In order to summarise the current status in relation to the characterisation of algorithm searching behaviour, a taxonomy and brief accompanying description of selected existing behavioural measures is proposed below.

2.3.1.1. Measures in decision space. These measures yield insight into the nature of an algorithm's search (i.e. distribution of solutions throughout the decision space) as it approaches convergence. Within this category, behavioural measures are based on statistics of the iteration-wise changes in the location of an algorithm's individual, or population of, solution(s) within the topology of the decision space. These measures include the following: (i) *distance measures*, which provide a direct measure of the degree to which an algorithm has converged, where measures describing the “spread of solutions” characterise properties of the hyper-sphere sub-set of the decision space within which an algorithm is searching (e.g. [Boese et al., 1994](#); [Zecchin et al., 2012](#)); (ii) *direction measures*, which provide a characterisation of the predominant “direction” along which an algorithm has focused its search, and have been used as part of updating strategies of various algorithms (e.g. [Kennedy and Eberhart, 1995](#); [Bayer and Finkel, 2007](#)); (iii) *historical*

frequency measures, which track solution points in decision space to provide a statistical characterisation of the history of an algorithm's generated solution properties, enabling the classification of solution attributes according to selection frequency statistics (Cunha and Ribeiro, 2004); (iv) *feasibility of solution measures*, which enable the search effort spent by an algorithm in the feasible/infeasible regions of the decision space (Zecchin et al., 2012) to be determined. This is of interest for many practical optimisation problems, where optimal solutions lie on the boundary of the feasible region in decision space.

2.3.1.2. Measures in objective space. These measures provide an indication of the quality of an algorithm's search, and yield insight into the most productive stages of the search. Measures that aim to characterise an algorithm's searching behaviour based on objective-space properties have been in broad use within both single- and multi-objective optimisation (e.g. tracking the iteration-best objective function value (e.g. Simpson et al., 1994)), or the evolution of the Pareto-front of non-dominated solutions (Zheng and Zecchin, 2014). Measures in the objective space have typically been used to compare an algorithm's end-of-run performance (see Section 2.4 for a broad overview of these performance metrics), whereas behavioural analysis focuses on the run-time tracking of these measures.

2.3.1.3. Algorithm dependent measures. Blum and Roli (2003) proposed an operator classification framework that categorizes operators on a continuum based on their propensity to intensify (e.g. exploit) or diversify (e.g. explore) an algorithm's search. Intensive procedures are deterministic and based solely on objective function information (i.e. steepest gradient searches), whereas diversification procedures utilise randomness and procedures not based on objective function information. Based on this classification, a resulting behavioural measure is the run-time search effort an algorithm devotes to either intensification or diversification (Birattari et al., 2006; Cunha and Ribeiro, 2004; Zeferino et al., 2009; Zufferey, 2012).

2.3.2. Research challenges and future directions

2.3.2.1. Relating behavioural measures to algorithm behaviour. The first open research question with regard to characterising an algorithm's search behaviour relates to whether there are appropriate metrics for characterising this behaviour. Broadly speaking, measures can be considered as being based on two dimensions, one related to space (the decision, and objective spaces) and the other to frequency. Concerning the spatial dimension, one open question is about the definition of topology (or distance) within the decision space. In contrast to the objective space, the decision space does not always have a clear topology, as many practical engineering problems contain mixed decision spaces (combinations of real and discrete decision dimensions) and complex decision tree structures (such as those arising from planning applications, e.g. Kasprzyk et al. (2009)), which require more general approaches to defining topology that are meaningful for behavioural analysis. A second issue concerning the spatial dimension is the characterisation of "spread". Decision space metrics have been used to characterise the degree of convergence of an algorithm, but not the nature of this convergence. In describing behaviour, measures that characterise the structure of the distribution of solutions through the decision space provide important information, such as knowing how clustered the solutions are (i.e. grouped about different extrema, or non-dominated decision-space points on the Pareto front), how many of these clusters exist, and the topological relationship between them. Concerning the frequency dimension, an open question is how to characterize the temporal aspects of the search

procedure as it evolves throughout the decision space. In fact, the number of times a solution is visited, the number of times a particular decision has been assigned to a particular variable, or the number of iterations concentrated in a given region of the decision space are all informative for characterizing algorithm behaviour and establishing convergence patterns.

2.3.2.2. Application/utilisation of behavioural measures. A second open research question concerns the application and utilisation of behavioural measures. Some of the research challenges and future directions associated with this are outlined below.

2.3.2.2.1. Building algorithm operator knowledge and algorithm development/design. Building on the intensification and diversification framework of Blum and Roli (2003), an important application of behavioural measures is to develop a detailed understanding of the impact of algorithm operators on searching behaviour, and in particular, the impact of the controlling parameters on these operators. A behavioural analysis framework provides the potential to gain a deeper understanding of the influence of controlling parameters and the sensitivity of an algorithm's convergence or explorative behaviour to changes in these parameters.

Given that behavioural measures provide run-time search statistics, they can be used as the basis for adaptive rules to select the most appropriate algorithm parameters or operators. For example, behavioural statistics could be used to track convergence and adaptively modify the values of controlling parameters to encourage exploration (or exploitation) and slow (or speed up) convergence, depending on the computational resources available. This issue is discussed further in Section 3.5.

Behavioural statistics could also facilitate the development of interactive algorithmic processes. There are recent efforts that specifically incorporate human interaction and visualization to characterize search behaviours (Babbar-Sebens and Minsker, 2008, 2010; Castelletti et al., 2010a; Kollat and Reed, 2007b; Lotov and Miettinen, 2008; Reed and Kollat, 2013; Singh et al., 2010). These studies outline techniques specifically developed for understanding, exploring and interacting with the EA search process. These visual methods overcome the difficulties associated with creating independent measures of search dynamics that capture both the dynamical evolution of decisions, as well as their single or multi-objective impacts on performance.

2.3.2.2.2. Interaction between behaviour and fitness landscape characteristics. As discussed in Section 2.2, the manifest behaviour of an algorithm results from an interaction between the algorithm's own operators and the fitness function. Consequently, developing understanding of the dependency of the algorithm's behaviour on the fitness function characteristics is important. This leads to the following questions: (1) Do different fitness functions induce different behavioural responses from an algorithm or algorithm operators? As mentioned in Section 2.2, the operator behavioural characterisations mentioned above cannot be decoupled from their dependency on the fitness function, and building an understanding of this dependency/interaction is important. (2) Are there behavioural characteristics that work more effectively for certain fitness function types than others? For example, for problem types that contain a "big valley" structure, initially exploitative behaviour followed by focused explorative behaviour could be more successful, however, the reverse may be true for problem types for which extrema are not tightly clustered within a sub-region of the decision space. A systematic evaluation of behavioural measures in relation to problem characteristics would also provide help with redesigning algorithms. It is important to overcome the classic calibration procedures that just establish links between parameters and performance statistics, which effectively consider the problem

type as a black box. Understanding what behavioural attributes are most successful for certain problems with certain characteristics is an important first step towards providing a guiding framework for using the most effective algorithms and algorithm operators for problems with certain characteristics.

2.4. Algorithm performance assessment

Algorithm performance criteria are defined in order to assess how well an algorithm or algorithm variant has performed on a particular problem, or compared with other algorithms. In addition, algorithm performance measures can be used to gain insight into algorithm searching behaviour (e.g. Cheung et al., 2003; Zecchin et al., 2012), as discussed in Section 2.3, or as a stopping criterion (see Section 3.6).

Algorithm performance is generally defined in terms of *effectiveness* (i.e. how close the solutions found by an algorithm are to the globally optimal solution, or the Pareto front, in the case of multi-objective problems) and *efficiency* (i.e. how quickly an algorithm finds its best solution(s)) (Silver, 2004). Due to the stochastic nature of EAs, other factors also have to be considered, including: (i) *reliability*, which measures the variability in effectiveness and efficiency for repeated implementations of an optimisation algorithm on the same problem; and (ii) *robustness*, which measures the sensitivity of an algorithm to parameters and problem characteristics.

2.4.1. Current status

2.4.1.1. Reliability. For single objective algorithms, reliability is represented by the average and standard deviation of the best solutions found in multiple runs, as well as the number of times the known best solution is found (Marchi et al., 2014). In multi-objective algorithms, reliability can be represented by the *empirical attainment function (EAF)*, which measures the probability of an algorithm finding a solution that is better than or equal to a certain solution in a single run. The EAF is able to identify the parts of a Pareto set that have not been explored by the algorithm (da Fonseca et al., 2001) and has been used by López-Ibáñez et al. (2010) and López-Ibáñez and Stützle (2012) as a graphical method to compare the reliability of different algorithms.

2.4.1.2. Robustness. When robustness is used to measure an algorithm's ability to solve different problems, it is often evaluated after the calibration of the algorithm parameters in an attempt to match the algorithm search to the problem characteristics. However, as parameter settings cannot always resolve issues related to the algorithm-problem interaction, and calibration is a time consuming process, research is moving towards the development of self-adaptive algorithms (discussed in section 3.5) and towards the concept of *controllability* (Hadka and Reed, 2012). A large controllability is a desirable property, as it means that an algorithm will be able to reach good results also when the parameters are modified.

2.4.1.3. Efficiency. As for many water resources problems the globally optimal solution(s) is generally unknown or unable to be located using particular algorithms, the allowable computational effort is often set *a priori* and the best solutions found within the given computational budget are compared.

2.4.1.4. Effectiveness. For single-objective algorithms, measuring algorithm effectiveness is relatively straightforward and is usually based on the best solution found. However, for multi-objective algorithms, the measurement of algorithm effectiveness is more problematic, as according to Zitzler et al. (2000), it is necessary to evaluate: (i) the distance of the final solutions obtained by an

algorithm from the true Pareto front; (ii) the coverage of the non-dominated space; and (iii) the extent of the non-dominated front. However, the quantification of these characteristics is not an easy task and it is recognised that there is no single measure that is able to correctly represent the performance of a set of non-dominated solutions (Deb and Jain, 2002).

Commonly used metrics measuring solution convergence of multi-objective algorithms are: (i) the *hypervolume* metric (Zitzler, 1999) or its variants, which calculate the hypervolume of the region dominated by a set of solutions (e.g. Kollat et al., 2008; Kollat and Reed, 2006; Reed et al., 2013; Zitzler et al., 2003) and are dependent on the selected reference point; (ii) the *error ratio*, which computes how many solutions in a non-dominated set do not belong to the Pareto front and is biased by population size (Knowles and Corne, 2002); (iii) the runtime *ϵ -performance metric* (Kollat et al., 2008), which is based on a similar idea to the error ratio, and is biased by the value of the user-defined tolerance ϵ used to compute the proportion of solutions closer than ϵ to the reference set; (iv) the *generational distance* (Van Veldhuizen and Lamont, 1998), which represents the average distance, or the normalised average distance as in Kollat et al. (2008) and Deb and Jain (2002), between a set of non-dominated solutions and the nearest solution on the Pareto set, and decreases when a non-dominated point far from the true Pareto front is added to a non-dominated set; (v) the *Maximum Pareto Front Error* (Van Veldhuizen, 1999), which measures the maximum distance between the Pareto front and the non-dominated solutions found, and the *additive ϵ -indicator* (Zitzler et al., 2003), which represents the distance required to translate the non-dominated set A so that all solutions in A dominate the solutions in a set B, and favours a set that has all solutions close to the true Pareto front, rather than a set that has all solutions on the Pareto front, except one that is further away compared with the previous set.

Commonly used measures of diversity for multi-objective algorithms include the Schott's *spacing metric* (reported in Knowles and Corne, 2002), the *Chi square deviation metrics* (Srinivas and Deb, 1994), the *diversity measure* (Deb and Jain, 2002) and the *Hole relative size* (HRS) (Collette and Siarry, 2005), which all measure the distribution of the solutions in the non-dominated set. In general, these metrics do not take into account that the solutions on the Pareto front may not be uniformly distributed.

There are other metrics, such as the *coverage metric C* (Zitzler and Thiele, 1999) and the *R metrics* described in Knowles and Corne (2002). However, these are not commonly applied, either because of issues with identifying the best non-dominated set or because of the computation time required (Deb and Jain, 2002).

Even though the quantitative metrics described above provide an indication of convergence and diversity, they can give misleading results, necessitating the use of visual comparison methods in many instances. One method used to visually assess algorithm performance is by a set of comparisons, each showing two of the objective function values of the solutions (e.g. Fu et al., 2012). Although this does not give a measure of how much an algorithm is better or worse than another, it highlights whether an algorithm is performing well in a particular region of the search space. The problem associated with this form of graphic evaluation is that it becomes more difficult when the number of objectives increases. For a relatively small number of objectives, colour coding and sizes of symbols can be used in order to assess solutions in higher dimensions (e.g. Fu et al., 2012; Kasprzyk et al., 2013).

2.4.2. Research challenges and future directions

Some of the research challenges and future directions associated with algorithm performance assessment include:

2.4.2.1. Development of accurate, platform-independent measures of efficiency. Measuring computational time using the number of function evaluations facilitates comparison of results obtained using algorithms run on computers with different computational power and by algorithms that use different population sizes and/or number of generations. However, there are still problems with comparing the run times of EAs with those of exact methods (e.g. linear programming) or measuring differences in the computational requirements of algorithm variants that incorporate different operators (e.g. local search), as algorithm operators can significantly increase overall run time without increasing the number of function evaluations. Consequently, there is a need to develop alternative measures of computational efficiency that are independent of the computational platform, but take all computational requirements into account.

2.4.2.2. Addressing issues associated with computational efficiency. Available computational power is currently a potential problem for two reasons: (i) testing algorithm performance requires multiple runs and (ii) some of the performance metrics are computationally expensive. Consequently, limitations in available computational resources could have a negative impact on the quality of the evaluation of algorithm performance. The magnitude of this problem could be reduced, although not completely resolved, by the introduction of more powerful computers and the use of some of the techniques for improving computational efficiency discussed in Sections 3.3 and 3.4.

2.4.2.3. Development of improved performance measures for multi-objective algorithms. For multi-objective algorithms, one of the major issues is that there is currently no metric (or set of metrics) that can completely describe the performance of an algorithm without biases. Therefore, there is still the need to confirm the results using some sort of visualization. Future research should therefore focus on: (i) developing a set of metrics that can reliably describe the performance of multi-objective algorithms without introducing biases caused by the choice of the reference point or the shape of the Pareto front; and (ii) improving the graphical visualization of multi-objective fronts.

3. Application to real-world problems

3.1. Overview

The application of EAs to real-world problems raises a number of methodological challenges, as illustrated in Fig. 3 and outlined below. Current progress and remaining challenges surrounding each of these issues are discussed in Sections 3.2–3.9.

When solving real-world problems, problem formulation becomes more complex, as the number of decision variables, decision variable choices and objectives is generally larger. In addition, the selection and numerical specification of the most appropriate decision variables, constraints and objectives becomes more difficult. This raises the question of how to best change the formulation of optimisation problems to cater to real-world problems (see Section 3.2).

As the number of decision variables and decision variable choices increases for real-world problems, so does the size of the search space. This makes it more difficult to find (near) globally optimal solutions and raises the question of what can be done to reduce the size of the search space, as “brute-force” optimisation approaches are unlikely to yield good results for search spaces of the size encountered (see Section 3.3).

Given that the search space is likely to be larger and the fitness landscape more complicated for real-world problems,

computational efficiency can become a significant problem. This is compounded by the fact that the simulation models used to evaluate objective functions and/or check constraints are generally much more computationally expensive, potentially making optimisation runtimes prohibitive or preventing the identification of near globally optimal solutions within available computational budgets. This raises the question of how computational efficiency can be increased for real-world problems, so that problems can be solved in a reasonable timeframe, while enabling near-globally optimal solutions to be found (see Section 3.4).

Due to larger search spaces, more rugged fitness landscapes, the need to deal with multiple objectives and decreased computational efficiency, it is likely to be more difficult to identify near globally optimal solutions for real-world problems. This raises the question of which searching mechanisms (e.g. algorithms and parameterisations) should be used for solving real-world problems (see Section 3.5).

As a result of the increased computational burden and the need to consider competing objectives when dealing with real-world problems, it is more difficult to determine when convergence has occurred and what an appropriate level of convergence is. This raises the question of what termination/convergence criteria are most appropriate for real-world problems (see Section 3.6).

The consideration of multiple objectives poses the potential for broader understanding of performance trade-offs, but requires commensurate effort in advancing the value of these results for aiding decision makers (Balling, 1999; Castelletti et al., 2010a; di Pierro et al., 2007; Kasprzyk et al., 2013; Kollat and Reed, 2007b; Lotov and Miettinen, 2008; Reed and Kollat, 2013; Woodruff et al., 2013). For example, for two- or three-dimensional problems, Pareto optimal trade-offs can be visualised relatively easily. However, higher dimensions require more advanced visual analytics. This raises the question of how to best convey results to decision makers and what role the use of EA optimisation plays in the overall decision-making process (see Section 3.7).

Uncertainty is a feature of real-world problems and affects a number of areas of the optimisation process, such as problem formulation, the evaluation of objectives and constraints, and computational efficiency. This raises the question of how to best consider uncertainty in the optimisation of real-world systems (see Section 3.8).

In order to use optimisation algorithms as part of decision-making approaches for real-world problems, they must be able to be linked with complex simulation models in an efficient and robust fashion. In addition, they must be able to cater to all aspects of the optimisation process in a user-friendly fashion, including problem formulation, decision-making and uncertainty. This raises the question of how to best achieve this (see Section 3.9).

3.2. Changes in problem formulation

This section focuses on how to formulate real-world problems to meet the needs of end users. Problem formulation involves the specification of objective functions, constraints and decision variables (Fig. 3). Objective functions are used to represent and quantify the broader problem objectives, such as defining a system's cost or engineering performance, constraints can set limits on performance, such as the minimum percentage of demand that must be satisfied, and decision variables encode the design parameters or policy levers that can change. When formulating real-world problems, emphasis should be placed on generating alternatives that differ in their decision variable performance. In other words, visualizing the decision variable space may be helpful to get “maximally different” alternatives (Brill et al., 1990). While, strictly speaking, problem formulation is concerned with the selection of

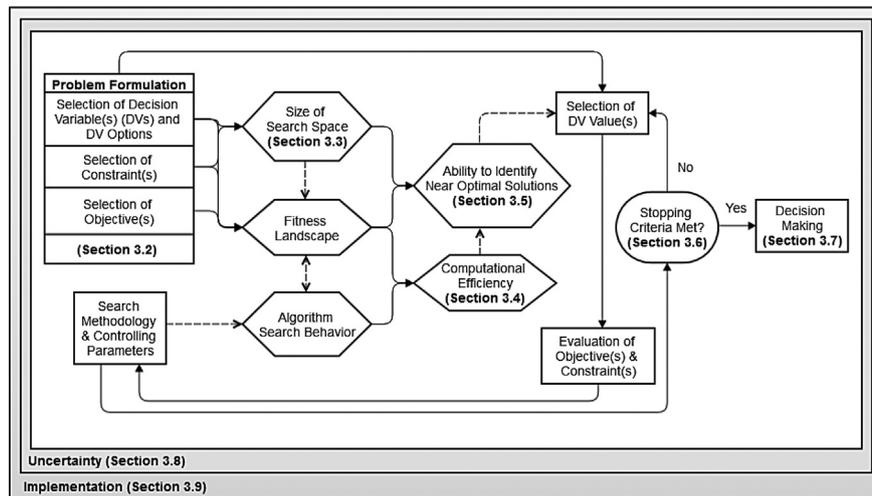


Fig. 3. Research challenges associated with solving real-world problems. The square shapes represent the steps in the EA process, the oval shape represents a decision point and the hexagonal shapes represent problem and algorithm characteristics resulting from choices made at different steps of the EA process.

objective functions, constraints and decision variables, this also has flow-on effects on other aspects of the optimisation process, such as the size of the search space, the characteristics of the fitness function, the ability to find near-optimal solutions and computational efficiency (Fig. 3).

It should also be noted that for many real-life problems, the distinction between objectives and constraints is not always clear cut and in some instances, objectives could become constraints, and vice versa. For “hard” limits, such as regulatory requirements, use of a constraint may be more appropriate. At the same time, having a constraint will not reward performance above and beyond this limit and by relaxing constraints, different levels of performance can be determined. When constraints are turned into objectives, the associated trade-offs with other objectives can also be explored.

While in most studies in the literature problem formulation is taken as a given, when solving real-world problems, this is not the case. Instead, problems need to be formulated between analysts, decision-makers and other stakeholders. While an ideal problem formulation would capture all decision makers’ goals, as well as each physical process relevant to the considered system, practical considerations can impose a number of limitations. For example, every objective function and constraint needs to be able to be evaluated, which is generally achieved with the aid of simulation models that link decision variable choices with the selected objective functions and constraints. However, such models might not be available, difficult to develop (especially for non-technical objectives) or computationally expensive. Therefore, analysts must have a greater focus on problem specification and the problem formulation itself. A number of the practical limitations identified by Rogers and Fiering (1986) in relation to systems analysis techniques are also relevant in this context. For example, the analyst may be forced, through rules or regulations, to use a specific simulation model. Similarly, they may be required to adopt certain decision variable definitions that may not be effective from the point of view of solving the problem. Additionally, with the increasing focus on social and environmental objectives, the problem can be challenging to represent in quantitative form.

3.2.1. Current status

Whilst a number of water resource optimisation studies have examined complex real-world problems, including consideration of

environmental and social water objectives (e.g. Becker et al., 2006; Wang et al., 2009; Yang and Cai, 2010; Yin et al., 2012), they are often limited by the ability to model these systems properly. In addition, the decision variables, objective functions and constraints are generally assumed to be known (e.g. Newman et al., 2014; Paton et al., 2014; Wu et al., 2013). However, some progress has been made in relation to the development of approaches that consider stakeholder input in the problem formulation phase. For example, participatory approaches in which problem formulations are constructed in collaboration with stakeholders have been advanced in a series of studies by Babbar-Sebens and Minsker (2008, 2010, 2012), Babbar-Sebens and Mukhopadhyay (2009), Piemonti et al. (2013), as well as Castelletti and Soncini-Sessa (2006). In addition, Singh et al. (2009, 2008, 2010) and Singh et al. (2013) present interactive optimization approaches that incorporate qualitative information into the problem formulation using stakeholder rankings of interim solutions. Similar ideas have been developed in Castelletti et al. (2010b, 2011) and “Problem Structuring Methods” can also help to implement traditional operations research approaches with diverse sets of stakeholders (Rosenhead, 1996).

Progress has also been made in relation to investigating the impact of the consideration of the number of objective functions. While real-life problems are generally characterised by multiple, competing objectives (e.g. social, environmental, technical, economic), different levels of aggregation of these can be used in the objective functions specified for the optimisation problem. If a single objective function is used, one optimal solution to the problem will be sought. However, some theoretical work has shown that it is difficult or impossible to create an internally consistent aggregation function to achieve this (Franssen, 2005). At the other end of the spectrum, there is an emerging paradigm for “many” objective analysis of more than four objectives (Fleming et al., 2005; Woodruff et al., 2013). However, the inclusion of too many objectives could prove problematic. For example, Teytaud (2007) discussed technical problems with including more than 10 objectives, and there are also perceptual limitations to considering this many performance criteria simultaneously.

3.2.2. Research challenges and future directions

3.2.2.1. Identification of impacts of simplifications and assumptions. Given the challenge of representing real world problems in an optimisation framework, greater emphasis is required to identify

the impacts of simplifications and assumptions, within both the model and the optimisation components, on the resulting solutions. An example of one such challenge is how the problem formulation changes over time. In this respect, the problem formulation itself can be seen as a type of optimisation problem. Three reasons for this are (i) that the needs of stakeholders change over time, (ii) technology (and therefore the suite of feasible alternatives) changes, and (iii) problem perception changes based on problem learning. In general, analysts should be concerned with how sensitive the ultimate selection of solutions is to the problem formulation considered.

3.2.2.2. *Mathematical representation of objectives and constraints.*

In addition to the difficulties associated with the articulation of appropriate objectives and constraints for real-world problems, analysts face the challenge of how to represent these as a set of mathematical equations. Although decision-makers may know their goals (for example, equity in trans-boundary management, or environmental concerns), it may be challenging to translate these into equations. The success of the optimisation paradigm depends on whether it is meaningful to map abstract or conceptual goals (such as improving ecosystem health) and other metadata into one or more numerical metrics, expressed as some combination of objectives and constraints. A practical limitation of such a mapping is the difficulty in comprehending the Pareto optimal trade-offs in many-objective problems and in arriving at a preferred solution. This is further addressed in Section 3.7 and in the multiple-criteria decision making literature (e.g. Hyde and Maier, 2006; Hyde et al., 2004, 2005). Additionally, there should be further work on linking the properties of algorithms to the number of objectives, especially with the increasing consideration of larger numbers of objectives when optimising real-world problems. This can be combined with approaches to reduce the number of objectives (e.g. Brockhoff and Zitzler, 2009; Giuliani et al., 2014a).

3.2.2.3. *Development of understanding of limitations and strategies for increasing relevance and credibility.*

In some regards, all formulations are based on scenarios or exogenous data that represent assumptions about the system which may not be within the control of the simulation and optimisation approach. As the optimisation outcomes are conditional on these scenarios, the practical relevance of the resulting solutions depends very much on how decision makers feel about the credibility of the scenarios or metadata used in the decision-making process (Aumann, 2011). This also has relevance to optimisation under uncertainty (see Section 3.8). Which simulation model is used is also a function of the computing technology used. Several years ago, high performance computing applications were much more difficult to use. However, high performance computing allows for longer function evaluation times (and thus larger simulation models to calculate objectives). Simulation model complexity may be able to be adjusted so as to be commensurate with computational demands. This can also facilitate changes to the problem formulation itself in the form of a learning feedback loop.

Another way to frame the argument is to ask: when is optimisation useful and not useful in solving real-world problems? There has been some discussion about the usefulness of optimisation in public sector planning, which predated EAs (see Liebman, 1976). Liebman argues that optimisation should be framed not as a process where the analyst gives “the” answer to the decision maker, but rather that optimisation facilitates learning about the problem. Even when optimisation is undertaken, the process could be considered to be fluid and iterative. For example, Becker et al. (2006), in an application of EAs to three real-world groundwater remediation sites, found that decision makers often asked for

modifications to the formulations (e.g. new constraints) after viewing initial results that gave new insights into the remediation systems.

As mentioned previously, EAs have bolstered our ability to solve problems that are more relevant to real-world systems. As this technical solution becomes more feasible, it is reasonable to expect more careful real-world scrutiny of the optimisation paradigm and its relevance to decision making. The challenge will be to develop a deeper understanding of the “sociology” of multi-objective optimisation in complex problems (i.e. how engineers, decision makers, policy makers and other stakeholders interact) to identify obstacles and devise corrective strategies.

3.3. *Reduction in size of search space*

As many real-world optimisation problems are defined on a large parameter search space, EAs may have difficulties finding solutions that are close to the global optimum. However, as a result of recent breakthroughs in algorithms for solving large optimisation problems, coupled with dramatic improvements in computing power, some algorithms are able to find solutions that are sufficiently close to the global optimum for practical purposes. Nevertheless, even in these situations, the required computational effort can be extremely large.

The above problem is a result of the so-called ‘curse of dimensionality’, where the search space grows exponentially with the growth in decision space dimension (Bellman, 1961), i.e. the number of decision variables or their precision. For example, if searching over a one-dimensional unit-interval space with precision of 10^{-2} , one needs 100 (or 10^2) sample points to visit each and every point. However, a search for a solution in a 10-dimensional space would require 10^{20} sample points to guarantee the same accuracy. It is, therefore, intuitive that a reduction in the size of a search space will lead to better computational efficiency (a lower number of function evaluations) and a greater chance of identifying higher quality solutions (i.e. closer to the global optimum/optima). However, for real-world problems, a reduction in the size of the search space generally results in its approximation, either because a number of decision variables have to be fixed prior to optimisation or because the nature of interactions among the decision variables precludes effective size reduction. This could potentially exclude the region that contains the true global optimum/optima, and thus reduce the quality of the solution(s) found. Consequently, some kind of reduction strategy for large search spaces is desirable, but with safeguards against missing the most promising regions of the space. As discussed in Section 3.4, strategies for increasing the computational efficiency of EAs can also be considered (separately or in conjunction with a reduction in the size of the search space) in order to address this problem.

3.3.1. *Current status*

A number of possible methods for reducing the size of a search space exist, including those based on *changes being made to the search space itself* (independently of the search algorithm used) or those based on *changes to the search algorithm* (i.e. incorporating some type of a problem-specific heuristic to reduce the size of the search space). A hybrid between the two approaches is also possible.

3.3.1.1. *Methods based on changes to the search space.*

Adaptive grid methods (where grid refers to the finite number of points in the search space) normally start out with an initial coarse grid with relatively few points and subsequently increase the number of points in the grid. Examples of such approaches are fixed-length encoding, such as Dynamic Parameter Encoding

(Schraudolph and Belew, 1992), or the use of variable-length chromosomes, as is the case with a Messy GA (Goldberg et al., 1989). Dynamic Parameter Encoding adaptively controls the mapping from fixed-length chromosomes to real values, such that at each subsequent iteration, the algorithm searches over a smaller search space. Ndiritu and Daniell (2001) used a similar approach to refine the search space of a GA used for rainfall-runoff model calibration.

Another approach to reducing the search space is based on a Delta coding GA (Whitley et al., 1991), for which each individual iteration corresponds to a single run of a GA with a changed mapping strategy. In particular, after each run of the GA, the algorithm is restarted with the coding for each decision variable representing a distance (or delta value) away from the most recent solution obtained in the previous GA run. The delta value is then changed to facilitate a refined search around the most recently found solution. Similar approaches have been used in water management practice by Afshar (2006, 2012). A Messy GA, on the other hand, uses variable-length strings that combine short, well-tested building blocks to form longer, more complex strings that increasingly cover all features of a problem under investigation. Examples of Messy GAs adapted and used for the design of water distribution networks are presented by Halhal et al. (1997), Walters et al. (1999) and Wu and Simpson (2001).

Another group of methods that change the search space itself uses specialised problem encoding to produce solutions that are always valid, thereby restricting the search space to the feasible region. However, these methods often require changes to the algorithm itself, as standard EA operators (e.g. crossover and/or mutation) may not be appropriate. Examples of such approaches are presented by Walters and Lohbeck (1993), Walters and Smith (1995) and Savic and Walters (1995). Alternatively, ant colony optimisation algorithms can be used to dynamically reduce the size of the search space during the optimisation using if-then rules for problems that can be represented in the form of a decision tree, where the selection of particular decision variable values affects the availability/feasibility of decision variable options at subsequent steps in the decision-making process (Foong et al., 2008a, 2008b; Szemis et al., 2013, 2012, 2014).

Probabilistic Model Building Genetic Algorithms denote a group of optimisation methods that search by identifying critical building blocks in the decision variable space (i.e. by identifying the most important decision variables and their relations). As a consequence, the search space is modified/reduced and hence the optimisation is performed in a more efficient manner. Several such methods, including the Hierarchical Bayesian Optimisation Algorithm and the Univariate Marginal Distribution Algorithm, have been used successfully for water distribution system optimisation by Olsson et al. (2009).

For problems that require evaluation of a computationally-intensive model with spatial dimensions, Babbar and Minsker (2006) and Sinha and Minsker (2007) developed multiscale GAs that reduce the search space by using coarse simulation model grids, periodically redirected using finer grids.

3.3.1.2. Methods based on changes to the search algorithm. Examples of methods based on changes to the search algorithm (using heuristic information/knowledge) include those that use some knowledge of the problem under consideration to modify the underlying search methodology. For example, Kadu et al. (2008), Zheng et al. (2011, 2014) used critical/shortest path methods to reduce the search space for the optimal design of water distribution networks.

Preconditioning is a technique that uses a set of known good solutions as starting points to improve the search process (Nicklow

et al., 2010). Although it does not directly change the optimisation algorithm, the results of preconditioning often improve the computational efficiency of the methodology. Examples from the field of the design of water distribution networks (Kang and Lansey, 2011; Keedwell and Khu, 2005), sewer networks (Guo et al., 2007), and groundwater remediation (Becker et al., 2006) use a set of domain specific rules to improve a solution before using it to seed a GA run. Alternative approaches are based on global sensitivity analysis (Fu et al., 2011), the use of engineering judgement and historical information (Pasha and Lansey, 2010), or graph decomposition methods.

The methodologies for reducing the search space presented above illustrate the approaches that have been tested and implemented in various studies. However, they also indicate what could be implemented in various EA problems, either as individual search space reduction methods, or as potential hybrids.

3.3.2. Research challenges and future directions

Despite advances in computing power and parallel computing, the need for reducing the size of the search space is still as acute as it was 30 years ago when EAs were in their infancy, as we attempt to solve larger and larger problems. Several research challenges and future directions emerge from the analysis of the current state of search space reduction methods:

3.3.2.1. Development of a generic EA methodology that will be independent of the problem under consideration. On one hand, a classical binary GA is an example of a generic EA methodology developed to tackle a wide variety of optimization problems. On the other hand, there are highly specialised EA methodologies that are developed to solve one specific type of problem. While the advantages of having a universally applicable methodology are obvious, the price to pay is often the loss of computational efficiency or poor quality results when applied to a problem with a large search space. For example, Guo et al. (2007), Kadu et al. (2008), Keedwell and Khu (2005) and Zheng et al. (2011, 2013, 2014) demonstrate that carefully structured problem specific EA implementations outperform their generic counterparts. From this standpoint, an important question to be answered is: Can more generic methods be developed (e.g. Rothlauf, 2011) to decompose large-scale problems into a series of approximately decoupled smaller problems (more amenable to practical optimisation), and recombine the information gained from the small problems to inform the solution process of the full-scale problem? It is likely that only a compromise is possible, where a balance between completely generic and problem-specific methodologies has to be obtained. A further question that needs an answer is: How would this methodology be different for continuous and discrete decision problems?

3.3.2.2. Development of improved preconditioning and iterative search space reduction methods. Various forms of preconditioning to improve search efficiency (e.g. by performing sensitivity analysis, identifying critical building blocks, and applying a deterministic heuristic) have been investigated in the literature. In addition to preconditioning an EA run, the process can be repeated with the algorithm being restarted at various times with each subsequent run using the most recent results (e.g. Afshar, 2006, 2012). It is, therefore, important to consider elements of a generic space reduction framework including, but not limited to, the best timing, discretization, and adaptive strategies (e.g. agent-based EAs, delta coding, etc.).

3.3.2.3. Development of theoretical and empirical analysis frameworks for space reduction strategies. Although theoretical analysis

of EAs has received increasing attention over the years, most studies in the available literature focus on a particular application within a narrow application domain (e.g. [Gibbs et al., 2008](#); [Simpson and Goldberg, 1994](#)). There is, therefore, much work to be done on investigating the theoretical underpinnings of space reduction methods for EAs. Similarly, research efforts aimed at investigating the computational efficiency of various search space reduction techniques in a systematic, empirical fashion, making use of rigorous statistical analyses, are needed.

3.4. Increase in computational efficiency

Whatever computational power is at one's disposal, there will always be a temptation to solve more complex problems faster. Apart from simply buying a faster computer, there are other ways to address this problem by performing model-based optimisation in a smarter way. The most important of these are: (i) surrogate modelling, (ii) the use of parallelization, (iii) population pre-processing, and (iv) using heuristic information to reduce the size of the search space, as discussed in Section 3.3. Areas where increased efficiency is especially important include (i) all problems where complex models are employed for fitness function estimation (e.g. two or three-dimensional finite element (volume) models used in hydrodynamics and groundwater), (ii) problems with extremely large search spaces (e.g. multi-reservoir optimisation problems), (iii) situations where very fast estimates of the optimal solution are needed (e.g. real-time control applications), and (iv) problems requiring Monte Carlo replication to make uncertainty "visible" to the optimizer (see also Section 3.8).

3.4.1. Current status

3.4.1.1. Surrogate models (SM). Surrogate modelling (also known as meta-modelling or emulation modelling) has been popular in applications where computationally intensive simulation models are used to estimate objective functions and constraints, as they mimic the behaviour of the simulation model as closely and consistently as possible, while being computationally cheap to evaluate.

One approach to emulating complex models is dynamic emulation modelling (or model reduction), where a high fidelity simulation model is replaced by a lower order dynamic model identified by selection or reduction (i.e. projection) of a subset of the original model variables ([Castelletti et al., 2012a](#)). This approach has been used, for example, for reducing the dimension of a groundwater model (see [McPhee and Yeh, 2008](#); [Siade et al., 2012](#)), for optimising groundwater pumping rates ([Ushijima and Yeh, 2013](#)) and for water quality remediation planning in lakes and reservoirs ([Castelletti et al., 2012b, 2014](#)). It should be noted that this approach is particularly useful for specific mathematical tasks requiring knowledge of system behaviour over the whole simulation horizon (e.g. optimal control).

When dealing with simulation-based optimisation, an alternative is to emulate the simulation model by developing a non-dynamic mapping (regression) of the complex model inputs into the outputs, namely objective functions and/or constraints ([Bieker et al., 2007](#); [Blanning, 1975](#); [Broad et al., 2005, 2010](#); [Gorissen et al., 2010](#); [Keating et al., 2010](#); [Kleijnen, 2009](#); [Laloy et al., 2013](#); [Queipo et al., 2005](#); [Yan and Minsker, 2006, 2011](#)). There are a number of reviews of the use of these techniques in optimisation, including [Jin \(2005\)](#) and [Knowles and Nakayama \(2008\)](#), as well as a dedicated edited volume on "Computational Intelligence in Expensive Optimization Problems" ([Tenne and Goh, 2010](#)) (in particular, reviews by [Shi and Rasheed \(2010\)](#) and [Santana-Quintero et al. \(2010\)](#) therein). In addition, [Razavi et al. \(2012\)](#) presented an extensive review of surrogate modelling in water resources and recent developments and applications to environmental systems are also presented in a

special issue on "Emulation techniques for the reduction and sensitivity analysis of complex environmental models" (see [Ratto et al., 2012](#)).

There are two problems to consider with SM: (i) choosing the best method for building a SM with data generated by the exact model (see [Santana-Quintero et al., 2010](#)) and (ii) choosing or developing MOEA schemes (or restructuring existing ones) that are able to use a SM instead of the exact model in the most efficient way. This is what is called "metamodel-enabled optimizers" by [Razavi et al. \(2012\)](#), and is referred to as SM-based optimisation (SMBO) algorithms (SMBOA) in this paper.

Historically, there has been an increase in sophistication of the algorithmic schemes used. Early papers on using SMs in water-related problems used a straightforward strategy for addressing the above issues. This includes sampling enough data by running the exact model and training a SM, and using it without change or further enhancement to estimate the objective function in an optimisation algorithm. Examples of this include [Solomatine and Torres \(1996\)](#), [Rao and Jamieson \(1997\)](#), [Maskey et al. \(2000\)](#) and [Khu et al. \(2004\)](#). A natural improvement to this approach is to use an iterative scheme of building a sequence of the progressively improving SMs for the current locality of search as optimisation progresses (thus forming a coordinated sequence of local models). This approach was possibly outlined first by [Nain and Deb \(2005\)](#). A variety of enhancements to this general approach have been proposed, including different sampling strategies ([Knowles, 2006](#)), dynamic algorithm control ([Gaspar-Cunha and Vieira, 2005](#)), and online retraining ([Bi and Dandy, 2013](#); [Xu et al., 2014](#); [Yan and Minsker, 2006, 2011](#)).

In practically all applications areas where optimisation is used with complex models, SMBO is employed as well. Except for the groundwater remediation design examples already noted, the approach has also been used extensively in water distribution system optimisation applications (e.g. [Behzadian et al., 2009](#); [Broad et al., 2005, 2010](#)) and in real-time control ([Hutton et al., 2014](#)).

As highlighted by [Razavi et al. \(2012\)](#), there are challenges and limitations to current surrogate modelling capabilities. Frequently, applications have been limited in the numbers of decisions (often < 10), types of decisions (typically real-valued), and numbers of objectives. A key concern in using this efficiency enhancement is to carefully ensure that these restrictions do not adversely limit the consideration of relevant problem formulations.

3.4.1.2. Parallel/cloud computing. Since most EAs can be easily parallelised, an obvious solution to increasing computational efficiency is to employ parallel computing devices. Lately, there are many options that are directly available in professional practice, including multiple cores, graphical processing units, accelerators and co-processors, and clusters of PCs (e.g. office clusters, or dedicated clusters), with supporting software like the MATLAB Parallel Computing Toolbox and cloud computing platforms like Amazon EC2. To make best use of this technology, computer codes have to be designed to enable synchronisation and exchange of information between different processors, and the design of algorithms has to take into account particular parallel implementations. Message-passing protocols, such as MPI (Message Passing Interface) or PVM are supported for a number of programming languages, and some languages explicitly support parallelisation, such as FORTRAN 2008 and higher level languages, such as Octave ([Vrugt et al., 2006](#)). In EA applications where function evaluations dominate total computation time, master-worker algorithms are appropriate and simple to implement ([Tang et al., 2007](#)). There are many examples of the successful use of the abovementioned approaches ([Mortazavi et al., 2012](#)).

3.4.2. Research challenges and future directions

Some of the research challenges and future directions associated with improving computational efficiency are outlined below.

3.4.2.1. Development of additional guidelines for SM development. Despite significant progress in the development of SMBO, as discussed above, there are still a number of research challenges that need to be addressed:

1. There are no rigorous guidelines in relation to the sampling strategies (i.e. running the exact model) required to enable adequate representation of the exact model by the SM. This is an issue both for SMs that are developed prior to optimisation and those developed progressively during optimisation. Over-sampling can have a significant negative impact on computational efficiency, and undersampling can impact accuracy. If information about the fitness function is available (see Section 2.2) sampling can be adaptive, with increased sampling in more “rugged” areas of the fitness function and reduced sampling in “smoother” regions of the fitness function, or by checking the rate of variation in the fitness function (e.g. by using the Lipschitz constant) and adjusting the sampling rate accordingly. A related approach is to begin with a small number of samples as a rough approximation and then “localize” sampling and perform more fine-grained sampling in promising regions of the search space (e.g. Xu et al., 2014; Yan and Minsker, 2006). Further research is needed to develop guidelines on how much and when to perform the coarse-vs. fine-grained sampling on a variety of applications.
2. Since inputs to data-driven SMs generally include the decision variables, and since these have a direct impact on the objective function and constraint values of interest, the number of inputs to SMs increases significantly for real-world problems. This makes SMs more difficult to develop and reduces their computational efficiency. Consequently, approaches need to be developed that enable only critical inputs to SMs to be identified, so that the number of inputs to SMs can be reduced. The use of sensitivity analysis, as was done by Fu et al. (2011), is one way of achieving this. Alternatively, the use of techniques for the selection of inputs to data-driven models (Bowden et al., 2005a, 2005b; Fernando et al., 2009; Galelli and Castelletti, 2013; Galelli et al., 2014; May et al., 2008a; May et al., 2008b) might be worthwhile. For example, these are at the core of selection-based dynamic emulation proposed by Castelletti et al. (2012b). An alternative approach to using data-driven SMs is to use projection-based model reduction techniques, as mentioned at the beginning of Section 3.4.1.

3.4.2.2. Addressing issues associated with SM errors. SMs are only an approximation and therefore subject to errors. This raises a number of interesting questions, including:

1. *What impact do errors in the SM have on the optimisation process?* This is particularly pertinent in the case where SMs are used to estimate constraints, as a slight over- or under-estimation of a constraint by the SM can render an infeasible solution feasible and vice versa, which could have a significant impact on the optimisation process. Some strategies for dealing with this issue have already been suggested (e.g. Broad et al., 2005; Broad et al., 2010), but these need to be generalised and tested in a systematic manner.
2. *Under what conditions are SMs better able to approximate simulation models?* It would be useful to have some information about the degree of difficulty with which different types of

surrogate modelling approaches (e.g. ANNs, splines, instance-based learning) are able to approximate simulation models with different properties (e.g. degree of nonlinearity, number and type of discontinuities). Some of the approaches discussed in Section 2.2 might be useful in this regard.

3.4.2.3. Overcoming technological issues associated with the implementation of parallel computing. In parallel computing, there are issues (many of them purely technological) that are awaiting solutions, including (i) ways to allow execution of the same code (model) on multiple cores of a single PC, (ii) better tools for programming graphical processing units, (iii) resolving issues of multiple software licenses for cloud solutions, and (iv) better understanding the effects of different parallelisation strategies (Cantu-Paz, 2000) on solution quality. However, the expectation is that in several years, the use of parallel computing in all of the forms mentioned above will be common practice, spanning possibly to smartphones used as processing nodes. This could support crowd-sourcing of both computational effort and stakeholder input on interim solutions.

3.5. Ability to find optimal solutions

Evolutionary algorithms are sometimes described as ‘off-the-shelf’ optimisation algorithms, capable of optimising any design or function. Whilst they are widely applicable to many optimisation problems, their ability to find optimal or near-optimal solutions often relies on the selection of appropriate algorithms and parameters. Selection of the most appropriate algorithm and algorithm parameterisation is particularly important when dealing with real-world problems, as complex simulation models are generally used for objective function and constraint evaluation, as discussed previously.

3.5.1. Current status

In order to determine the most appropriate *parameter settings* of EAs, many papers have focussed on the use of extensive trials with different parameterisations. In addition, there is a great deal of literature on the more sophisticated idea of automating the tuning and selection of algorithm parameters. Example algorithms of this type include adaptive mutation operators (e.g. Serpell and Smith, 2010), which can adapt to a number of signals during the run: (i) time-variant, where the parameter changes according to the length of time the optimisation has been running; (ii) progress-variant, where the parameter is tuned according to the level of success attained by the algorithm and (iii) operator-improvement-variant (Hadka and Reed, 2013), where the probability of applying an operator is related to its previous success in generating good solutions. Alternatively, appropriate parameters can be selected based on underlying theory (e.g. Gibbs et al., 2008; Gibbs et al., 2010; Gopalakrishnan et al., 2003; Reed et al., 2007, 2000) or the properties of the fitness function (e.g. Gibbs et al., 2011).

In order to identify the best *algorithm*, significant research effort has been directed towards the development of a single universal genetic operator for population evolution that is always efficient for a diverse set of optimisation problems. Exceptions include self-adaptive or memetic algorithms combining global and local search in an iterative fashion. Reliance on a single biological model of natural selection and adaptation presumes that a single method exists that can efficiently evolve a population of potential solutions through the parameter space and work well for a diverse set of problems. However, existing theory and numerical benchmark experiments have demonstrated that this is not possible. This is because the characteristics of the fitness function can vary

considerably between different optimisation problems, as discussed in Section 2.2, and perhaps more importantly, can change *en route* to the global optimal solution.

In recognition of this fact, a number of self-tuning and self-organising optimisation algorithms have been developed (Hadka and Reed, 2013; McClymont et al., 2013; Vrugt and Robinson, 2007; Vrugt et al., 2009). For example, selective hyperheuristics (McClymont et al., 2013) control movements in the search space by determining the order of application of a set of lower level heuristics, effectively optimising the algorithm whilst optimising the problem. Multi-method search (Vrugt and Robinson, 2007) load-balances a set of global optimisation algorithms and problem-specific heuristics by assigning computational time to the algorithm that has generated the most effective solutions. These methods have begun to demonstrate that the precise configurations of algorithms and the way in which they search the search space can be adapted online during the optimisation process, and that this adaptation can be used to overcome difficult areas of the search space and to create custom algorithms for a particular problem instance.

3.5.2. Research challenges and future directions

3.5.2.1. Selecting the best algorithm and algorithm parameterisation. The searching methods described in Section 3.5.1 each adapt to the problem to which they are applied. The selection of the best method to achieve the required optimisation performance is therefore perhaps less important than with other algorithms. However, these methods can be placed on a continuum in terms of flexibility and computational complexity, at least from a theoretical perspective. Adaptive operator methods are the least flexible in that they are restricted to operating with a single algorithm type, but are the least computationally complex, as the only additional search requirement is the discovery or modification of application probability or probabilities. Multi-method search is more flexible in that the method selects an algorithm or combination of algorithms that performs best on a particular problem instance, but there is additional computational load in searching algorithm space in addition to the search space. Selective hyperheuristics are the most flexible in that they can create algorithms that transcend traditional algorithm boundaries, but there is an inevitable additional computational penalty in simultaneously searching both the operator sequence and problem space. However, this is a rather idealised scenario and in reality, the lines between these methods are rather blurred, depending on their exact formulation and parameter settings.

While these algorithms are able to adapt to various problem types and are therefore more widely applicable, deploying these algorithms will require careful consideration of a new set of search parameters relating to the search of the ‘algorithm’ space. Some questions that require an answer in relation to this are (i) Which type of algorithm (e.g. adaptive operator methods, multi-search methods, hyperheuristics) is most appropriate for a particular application?; (ii) Which operators/heuristics/algorithms should be included in the process? Larger numbers increase flexibility, but also increase the search space and longer runs are required to gain enough statistical information on each operator, heuristic, and algorithm; (iii) How best should we prioritise one operator, heuristic, or algorithm over another? The method chosen must ensure that it is searching the optimisation space effectively and many of the same challenges exist with searching this space as the actual problem space; and (iv) What information should be collected from the optimisation? Optimisation progress can be measured in many ways, particularly when many objectives are considered (see Sections 2.3 and 2.4). Determining the markers of performance will be

crucial to the correct weighting of operators, heuristics, and algorithms.

In relation to point (iii) above, one way of improving the performance of multi-search methods might be to increase the continuity in the way information from different algorithms is used. At present, multi-method optimisers generally use a discrete set of search algorithms to evolve the population to the global optimum. The number of points each constituent algorithm is allowed to create at the current generation depends on its (immediate) past reproductive success. Further improvements to the search efficiency could probably be made if such switching between algorithms is replaced with a more continuous update rule.

In relation to point (iv) above, a potential area of improvement is the utilisation of fitness landscape information (see Section 2.2) to inform algorithm parameterisation. At present, reproductive success is used as the main indicator of performance, resulting in an indirect use of fitness landscape information to iteratively guide the selection of an appropriate optimisation algorithm at different stages of the search. However, algorithm efficiency could potentially be enhanced significantly if more explicit knowledge of the fitness landscape is being used (see Sections 2.2 and 2.3). If we could quantitatively summarize, in one or multiple diagnostic metrics, the shape and geometry of the local (and global) fitness landscape, then, for multi-method algorithms, for example, this information could be used to select an algorithm that is known to exhibit the most efficient searching behaviour under those conditions. For example, if such measure(s) tell(s) us that the (local) response surface is rather deterministic, then a gradient-based local search algorithm will suffice and be most efficient. On the contrary, if the (set of) diagnostic metrics indicate the presence of a rather chaotic and non-ideal fitness landscape with numerous local optima, stochastic search methods, such as CMA-ES and simulated annealing, would be preferred. Thus, summary diagnostics of the fitness landscape will help decide which algorithm to use at what stage of the search. Such procedures would mark a major advance in the field of optimisation, and provide much better insights into the peculiarities of a given optimisation problem (see Sections 2.2 and 2.3). However, accurate calculation of appropriate fitness landscape characteristics in high dimensions is difficult and computationally demanding. Consequently, as discussed in Section 2.2, there is a need to determine whether there is value in following this path.

3.5.2.2. Efficient implementation of algorithms. Techniques such as multi-method search and hyperheuristics offer parallelism not just in computation, but in the optimisation approach. One of the key challenges that awaits researchers is the deployment of these algorithms on hardware platforms with very different characteristics (see Section 3.4). The rise of general purpose GPU processing heralds a new type of computation where many, relatively low-powered processing units are tightly integrated. As has been well documented elsewhere, GPUs represent the best trade-off between cost and performance in current hardware and so the adaptive methods described here will need to embrace this development (Tsutsui and Collet, 2013). There are challenges, of course, as constructing an optimisation approach for a GPU is very different to writing one for a CPU, multi-core CPU or even an MPI implementation. This is because GPUs have relatively small amounts of onboard memory and even smaller amounts of local ‘cache’ memory, which needs to be utilised efficiently to keep the (often thousands) of processor cores busy. However, the largest task is to determine how to best divide an adaptive optimisation process running on a water resources problem into many thousands of simultaneous parts, and one suspects that this will ultimately be best achieved through some automated process.

3.6. Stopping/convergence criteria

The purpose of algorithm stopping/convergence criteria is to decide when an EA should terminate. The criteria should ensure that premature termination does not occur, while also avoiding unnecessary computations. For well posed operations research problems, such as linear and nonlinear (e.g. gradient methods) programming models, convergence criteria are well defined, as the algorithm's analytical properties are quantified, and can be further translated into analytical optimality conditions. However, evolutionary algorithms are usually invoked in those instances in which conventional non-linear programming techniques are likely to fail. In such cases, approximate stopping criteria are needed, as the problems have no defined properties (e.g. convexity) to rely on for extracting conditions under which an optimal solution is attained.

3.6.1. Current status

Zielinski et al. (2005) provided the following classification of criteria for EA convergence/termination:

1. *Exhaustion-based criteria*: reaching a certain number of objective function evaluations or number of generations. These are the most common criteria in EAs and depend on a time constraint limit or CPU budget.
2. *Reference criteria*: the algorithm terminates when a certain percentage of the population has converged.
3. *Improvement-based criteria*: when improvements in the objective function continue to be small for some time, the optimisation algorithm is terminated, using criteria such as: (i) the best/average objective function value improvement is below a threshold for a number of generations, or (ii) no objective function improvement for a specified number of generations.
4. *Movement-based criteria*: changes in the population with respect to the average objective function value or their parameters are below a threshold for a number of generations.
5. *Distribution-based criteria*: using "distances" as criteria for convergence, such as checking whether the maximum distance from every vector to the best population vector or the standard deviation of the vectors is below a threshold.
6. *Combined criteria*: stopping criteria combinations, such as: if the average population/objective function improvement is below a threshold for a number of generations, check if the maximum distance is below a threshold.

Zielinski and Laur (2007) evaluated the performance of several stopping criteria that react adaptively to the state of the optimisation run. The examination was done on the basis of a constrained single-objective power allocation problem using a particle swarm optimisation (PSO) algorithm. Marti et al. (2007, 2009) developed a stopping criterion for single and multiobjective optimisation problems which combines the mutual domination rate (MDR) improvement indicator, along with a simplified Kalman filter that is used as an evidence-gathering process. The MDR is aimed at tracking the progress of the optimisation with low computational cost and thus at solving single or multiobjective optimisation problems efficiently. Studniarski (2010) derived a stopping criterion for a general Markov chain model of a multiobjective genetic algorithm. Through establishing an upper bound, a stopping criterion is derived which guarantees that at least one minimal element is a member of the last generated population. Bhandari et al. (2012) suggested that the variance of the best fitness values obtained in the iterations of a genetic algorithm, coupled with elitism, should be considered as an improved termination criterion for the optimisation process.

3.6.2. Research challenges and future direction

The research literature on stopping/convergence measures for EAs typically consists of three stages: (1) review of existing criteria, (2) suggestion of new, improved measures, and (3) testing of the suggested measures on given test functions and/or simple case studies.

Through stage (3) of the above process, the efficiency/advantages of the proposed measure(s) are aimed to be "proved". Stage (3), however, does not hold a generic proof of the suggested measure's success, as this is case dependent. The initial question of termination selection criteria for a given problem (which is likely to be different than any of those presented in the literature) is thus left unresolved. The advantage, however, of new criteria is in their diversity.

Future directions for the development of stopping/convergence criteria for EAs can consist of the following: (1) to mathematically relate rigorous properties of the complexity of different problems to selected stopping criteria (e.g. model size, number and type of equations and decision variables, level of nonlinearity), (2) to link stopping conditions to the parameters of EAs (e.g. for a GA, types of selection, crossover, and mutation), (3) to establish an empirical library of types of problems or problems with certain properties of the fitness landscape (see Section 2.2) versus successful convergence measures. The success of any new termination measure should be tested against the traditional default stopping condition of a maximum number of generations or maximum CPU/execution time.

The methods listed above can be combined with a qualitative, visualization-based approach to setting stopping criteria. Interactive visualizations (Kollat and Reed, 2007a) of the search process as a function of increasing generations can be employed to indicate search progress. If the objective function (or Pareto approximation set) is not improving, this information can supplement the quantitative stopping criteria to add information for the analyst.

In general terms, it is not expected that there will be an *a priori* guarantee that any particular criterion will work better than another for a given problem. Testing of different options will always be a necessity. However, general guidance on better convergence measures for categorical problems should and could be provided. This is particularly the case for multi/many objective problems, for which the measurement of performance and convergence is particularly difficult (see Section 2.4).

3.7. Decision-making

Ultimately, optimisation is a tool used to support decision-making and improve problem understanding. Making good decisions is difficult, highly subjective, and dependent upon stakeholder views. Almost all water resources problems are characterised by multiple conflicting performance criteria, compounded with conflicting views on the definition of what makes a "good" decision. This speaks to the importance of this issue. Involving decision makers in the optimisation process is therefore important to enhance their ability to ultimately make sound decisions.

3.7.1. Current status

Whilst optimisation is well recognised in research as being a useful tool to assist in water resource management decisions (e.g. Nicklow et al., 2010), there has been a lack of uptake by decision makers (Jones et al., 2002; Oliveira and Loucks, 1997). However, there is increasing interest in adopting such tools, particularly in urban water planning, such as discussed in a recent planning exercise at Colorado Springs Utilities in the USA (Basdekas, 2014). This

interest highlights the need for providing accessible optimisation decision making tools and support.

As with any decision making process, the way in which optimisation is applied is dependent on decision maker objectives. As discussed in Section 3.2, “real-world” problems are rarely defined by a single performance measure. Instead, they are best solved using multiple performance criteria (referred to as “many-objective” problems (Purshouse and Fleming, 2007)) that capture a broad range of system properties and stakeholder preferences, ultimately requiring that the decision maker choose from a suite of Pareto optimal design possibilities.

Explicitly incorporating decision maker preferences in optimisation generally falls under *a priori* approaches or *a posteriori* approaches. *A priori* methods seek to model decision maker preferences prior to searching for optimal designs (Coello et al., 2007), while *a posteriori* methods allow those preferences to be expressed following optimisation. A classic example of an *a priori* approach is multi-attribute utility analysis (MAUA) (Keeney and Raiffa, 1976, 1993), whereby preference structure is assigned to the various performance criteria by the decision maker prior to optimisation, and the resulting solutions to the optimisation are reflective of this preference structure. However, the reality that decision makers often “...don't know what they want until they know what they can get...” (Loucks, 2012) leads to interesting negative implications of this and other *a priori* methods.

However, several criticisms exist regarding *a posteriori* approaches to decision making (Coello et al., 2007; Haimes, 1998; Zeleny, 2005). One of these is that it is mathematically difficult and hence, computationally expensive to find the Pareto optimal solution set. This is currently being addressed through the development of numerous multi-objective evolutionary algorithms that are faster, more reliable, and easier to use than many of their predecessors (e.g. the Borg MOEA (Hadka and Reed, 2013)), as well as some of the other approaches for increasing computational efficiency discussed in Sections 3.3 and 3.4. Some interactive approaches (e.g. Hwang et al., 1979) have been developed in order to interactively explore the Pareto space without having to fully compute it in advance, thus mitigating the associated computational burden (e.g. Deb et al., 2006; Thiele et al., 2009). The complexity and high number of questions to be posed to the stakeholders remain an unsolved problem (Larichev, 1992).

An additional criticism is that the large Pareto sets resulting from the optimisation process tend to be challenging to visualise and interpret. However, several recent studies have shown that effective use of visual decision support frameworks can alleviate this by providing tools that assist in finding and focussing on designs representing the best compromise between all objectives and stakeholders (Kollat and Reed, 2007b; Kollat et al., 2011; Reed and Kollat, 2012).

3.7.2. Research challenges and future directions

Research challenges related to optimisation for decision making include the need for technological advances in algorithms and decision support tools; improved demonstration of how optimisation can be applied to real-world problems as part of a wider decision making process; and improved communication and access to information on how to apply optimisation.

3.7.2.1. Advances in algorithms and decision support tools. Although water resources applications are inherently multi-objective, both the research community and decision makers are only in the early stages of seriously considering truly multi-objective approaches to problem solving. Improvements to decision making software tools and techniques are needed to advance the evolutionary optimisation field in a way that emphasises

computational efficiency (see Section 3.4), and more fully incorporates the decision maker into the optimisation process. Future research should emphasise the development of software and techniques that enable real-time visualisation of the evolutionary optimisation process (e.g. the evolving Pareto front, decisions, constraint violations, etc.). Technological advances in this area would ultimately allow the decision maker to play a more active role in “steering” the optimisation process. Steering techniques will help enable so-called “de Novo” planning approaches (Kasprzyk et al., 2012; Zeleny, 2005), where the selection of decision variables, objectives, and constraints could be updated in real-time to allow for decision makers to incorporate real-time learning regarding problem structure and potential solutions.

The ability to visualise the shapes of trade-offs between multiple conflicting performance criteria and to quickly identify points of diminishing return within the trade-offs will be important to understanding the complex relationships that typically exist between objectives (Branke et al., 2008; Kollat and Reed, 2007b; Lotov et al., 2005; Madetoja et al., 2008). Tools which permit the decision maker to interactively “brush” or filter designs from view (Inselberg, 1997) according to objective value, decisions, constraint violations, and other variables are a must for rapidly distilling large Pareto optimal design sets into smaller, more manageable subsets of interest. Pareto sorting tools that allow the user to identify optimal designs using smaller, simpler sub-problems within the larger multi-objective optimisation problem (Kollat and Reed, 2007b; Kollat et al., 2011; Reed and Kollat, 2012) should be incorporated directly in these frameworks, as should techniques that allow the user to “thin” the Pareto set based on meaningful precision (Kollat and Reed, 2007a; Laumanns et al., 2002). Additionally, advances toward adaptive software frameworks that more fully integrate the decision maker into the modelling, optimisation, and the decision making process as a continual feedback loop of learning and understanding will ultimately lead to better decision making (Roy, 1971; Tsoukias, 2008; Woodruff et al., 2013).

3.7.2.2. Improving optimisation and decision making for real-world problems. Demonstrating the use of optimisation as a complementary tool in a set of wider management practices will assist in its uptake by decision makers. Optimisation should be viewed as a way of exploring possible management alternatives for further investigation, rather than focussing solely on finding optimal solutions, and enabling integration of stakeholder engagement and expert input.

3.7.2.3. Improved communication and access to information on using optimisation. Additional communication and support for applying optimisation, particularly for complex problems, is key to its uptake by decision makers. This includes access to optimisation algorithms and information on their suitability for different types of problems (see Sections 2.2 and 2.3), improving optimisation and decision support training, and developing more automated parameter and algorithm selection approaches that require less expertise to use effectively (see Section 3.5). It also includes guidance around formulating optimisation problems for complex systems representing social and environmental objectives (see Section 3.2).

In summary, improvements in decision support tools, as well as more integrated approaches to applying optimisation within existing decision making structures, are critical to advancing this field. The ability to optimise more complex problems through the use of EAs requires greater focus on providing guidance around representing these problems, and understanding the influence of assumptions and uncertainties on the outcomes.

3.8. Incorporation of uncertainty

Uncertainty is present in all aspects of water resources management (WRM) optimisation, from problem formulation to solutions obtained (see Fig. 3) and a lot of work has been published in recent years on uncertainty quantification in the water resources literature (Beven, 2006; Gupta et al., 2008; Kavetski et al., 2006a,b; Renard et al., 2011; Vrugt et al., 2005, 2008). One general source of uncertainty stems from imperfect knowledge about socioeconomic drivers in water systems, such as future water demand and population growth, and the extent to which urbanisation will occur. Hydrologic uncertainties are also important; there may be unknown water availability at sources and uncertain reservoir inflow due to natural climate variability, and modifications to the hydrologic cycle due to future climate change or changes in land use. Another important source of uncertainty deals with changing system dynamics, including the ageing of distribution and reclamation systems, and the uncertain stage storage characteristics of reservoirs due to sedimentation. Other uncertainties exist as well (e.g. future directions of socioeconomic development, legal/regulative framework, etc.). When dealing with real-world problems, it is important to consider these uncertainties in the optimisation process.

3.8.1. Current status

Many sources of uncertainty are complex in nature and interlinked with each other, and hence difficult to characterise. It is therefore not surprising that in many of the past WRM optimisation approaches, the aforementioned uncertainties were not considered, or that this was done in a simplified, implicit way (e.g. via a small number of future scenarios). The WRM optimisation approaches where uncertainties are handled in a more explicit way have appeared relatively recently (e.g. Cui and Kuczera, 2005; Deb and Gupta, 2006; Fu and Kapelan, 2011; Gopalakrishnan et al., 2003; Kasprzyk et al., 2009; Kourakos and Mantoglou, 2008; Labadie, 2004; Mortazavi et al., 2012; Singh and Minsker, 2008; Smalley et al., 2000; Tolson et al., 2004; Vasquez et al., 2000).

In these approaches, typically only a small number of key sources of uncertainty that are of primary interest for the WRM problem being analysed are identified and explicitly characterised. This is usually done by using stochastic models and scenarios, although recent examples of alternative approaches exist (e.g. Kasprzyk et al., 2013; Korteling et al., 2013; Matrosov et al., 2013; Mortazavi et al., 2013b). The observed data required to characterise the above uncertainties are often not available (Jakeman and Letcher, 2003) and hence subjective probabilities are frequently assigned. In addition, the selected uncertain variables are sometimes modelled in a very simplified way, without giving necessary consideration to spatial and temporal correlations, despite the fact that these can have important implications (Kapelán et al., 2005). In addition, the structural uncertainty of simulation models is often neglected. Finally, it should be noted that under conditions of deep/severe uncertainty (Lempert, 2002) that exist in real-life WRM problems, for some sources of uncertainty (e.g. future changes in regulative frameworks), pre-characterisation may lead to suboptimal solutions should future outcomes turn out to be different than originally anticipated.

In terms of WRM optimisation problem formulation and the corresponding solution methods used, existing approaches can be broadly classified into implicit and explicit stochastic optimisation approaches (Labadie, 2004). Implicit approaches use some sampling method (e.g. Monte Carlo) to generate data for stochastic variables of interest (e.g. reservoir inflow) before some deterministic optimisation method (e.g. EA or other, see section 3.5) is used to determine the corresponding optimal WRM configuration and/

or operational policy. Implicit WRM approaches appeared earlier and hence were typically based on conventional optimisation methods (Hiew et al., 1989; Hsu and Cheng, 2002; Martin, 1983; Tejada-Guibert et al., 1990; Yakowitz, 1982; Young, 1967). The deterministic optimisation process is repeated for several different instances of uncertain variables. Whilst this is appealing for a number of reasons (e.g. use of proven and computationally less demanding deterministic optimisation methods), the implicit approach requires results from multiple optimisation runs to be merged, which is not straightforward and is likely to lead to sub-optimal solutions that are not sufficiently robust.

In the explicit stochastic optimisation approach, the identified uncertainties in the WRM problem analysed are characterised and then quantified in an explicit manner during the (stochastic) optimisation process. Early explicit WRM approaches were based on conventional optimisation methods, such as Stochastic Linear Programming (Jacobs et al., 1995) and Stochastic Dynamic Programming (Tejada-Guibert et al., 1995). More recently, different EAs were used (e.g. Cui and Kuczera, 2005; Deb and Gupta, 2006; Gopalakrishnan et al., 2003; Kasprzyk et al., 2009; Kourakos and Mantoglou, 2008; Singh and Minsker, 2008; Smalley et al., 2000; Tolson et al., 2004; Vasquez et al., 2000). The original deterministic optimisation problem objectives are modified, typically by: (a) introducing some measure of robustness (or risk, resilience, reliability, etc.) as an indicator of water resource system performance and/or (b) applying some statistic to the total costs (e.g. averaging them). Whilst appealing from the uncertainty handling point of view, the explicit approach suffers from long computational times. Different remedies have been developed in the past to deal with this issue (see Section 3.4). Having said this, none of the existing approaches offer a generic and holistic solution for effective and efficient uncertainty propagation during the optimisation process. Regarding the decision variables used in optimisation-based WRM approaches, these are almost exclusively modelled as deterministic. However, this often results in rigid, precautionary strategies that may not be sufficiently flexible to adapt to uncertain future changes (Basupi and Kapelan; Giuliani et al., 2014b; Mortazavi et al., 2013a; Woodward et al., 2014).

In summary, uncertainty is unavoidable in WRM problems, yet decisions need to be made. While uncertainty has to be managed, optimisation offers the prospect of finding creative solutions that best manage this uncertainty. So the overall challenge is to ensure that the optimisation can “see” all significant uncertainties that affect performance, to specify objectives that not only seek efficiency (in its widest sense), but also robustness (or the ability to cope if future trajectories of the system are unfavourable), and to ensure decisions are framed to allow flexibility (or the ability to adapt to changing circumstances). Multi-stage decision methods (Beh et al., 2014, 2011; Watkins and McKinney, 1997) can provide some flexibility, but more flexible and adaptive approaches are needed.

3.8.2. Research challenges and future directions

Given the current status outlined in the previous section, there are several key research challenges for future work in this area:

3.8.2.1. Development of improved characterisation of WRM related sources of uncertainties. While key sources of uncertainties in WRM problems have been identified, a critical remaining challenge is to characterise these more accurately. In the cases where this is possible, this means collecting additional observed data that are required to, for example, better understand and define the uncertainty of various WRM model parameters (e.g. future system demand, etc.). In cases where the required observed data cannot be collected in sufficient quantities, identifying which data would

provide the greatest reduction in uncertainty becomes critical. If, however, the particular source of uncertainty is so deep that the required data can never be collected, then it should not be characterised at all and alternative WRM frameworks need to be developed (see below). Finally, WRM model (i.e. structural) uncertainty needs to be better understood and quantified in the future.

3.8.2.2. Development of improved problem formulations under uncertainty. Unless the key sources of uncertainty are made “visible” to the optimiser, the optimiser will have an overinflated sense of control and produce solutions that are likely to be suboptimal and possibly reckless – decision makers would rightly shun such solutions. The challenge is threefold: (i) how to best “expose” the different types of uncertainty to the optimiser; (ii) how to formulate the problem and objectives so that the optimiser uses information about uncertainty in a manner that reflects the needs of the decision maker; and (iii) how to do this in a computationally practicable manner.

Alternative problem formulations that make better use of the uncertainty embedded in the optimisation formulation include: (i) Increasing the number of diverse objectives addressing different aspects of the WRM problem analysed (e.g. environmental impacts/costs, health risks, greenhouse gas emissions, social acceptance); (ii) Incorporating risk, resilience and robustness based system performance indicators to ensure development of solutions with increased system redundancy (Grayman et al., 2012). This, in turn, will enable water resources systems to cope better with the full range of uncertainties (i.e. both 'known unknowns' and 'unknown unknowns' (Maskey et al., 2000)), by explicitly taking into account both the likelihood of possible failures and the consequences of these failures; (iii) Improved representation and evaluation of potential solutions based on the concepts of flexible design/real options (Dobes, 2008; Zhang and Babovic, 2012) and similar principles leading to solutions that are able to more easily/better adapt to future changes in the natural environment and the socio-economic system; (iv) Alternative frameworks that acknowledge the existence of deep/severe uncertainties and (v) Improved linking of stakeholder preferences with optimisation and decision making processes under uncertainty (see also Section 3.7).

3.8.2.3. Development of improved uncertain problem solution methods. Existing optimisation approaches should be improved via the development of improved uncertainty propagation/quantification methods. Most existing uncertainty propagation methods used in WRM are transferred directly from general statistical theory. As such, they are applicable to uncertainty sources characterised only in a conventional way (e.g. using PDFs). There is a definite need to develop new, WRM specific methods that can propagate uncertainties characterised in a number of different ways (see above). Also, this needs to be done in a computationally efficient and accurate way, which is a difficult task.

Another topic to explore in this context is to develop improved surrogate (or meta) models for more accurate uncertainty quantification in the part of the objective space where it matters (see Section 3.4). A need also exists to develop guidelines on how to (with minimum computational effort) select the best sample size for optimisation under uncertainty, given a range of possible metrics (e.g. risk based, robustness based, etc.). For example, Bayer et al. (2010) investigated the sampling of critical model realizations in order to reduce computational complexity. In addition to the above, new, more effective and efficient optimisation methods should be developed for solving WRM problems, where uncertainty does not hinder elitism and improvement.

3.9. Algorithm implementation

There are numerous options for implementing a search algorithm in computer code and linking or integrating it with a given environmental simulation model (see Fig. 3). Generally speaking, there is a trade-off between ease-of-implementation and computational performance. Codes that emphasize decoupling EAs and simulation models are often quite *flexible* (i.e. easily adapted to other problems and/or adopted by other research groups). Conversely, careful algorithm implementation and/or intelligent model integration can potentially yield order-of-magnitude improvements in computational performance (see Section 3.4 for a discussion of performance issues). However, realising such gains can require extensive multi-disciplinary expertise and the resulting implementation may not be flexible.

3.9.1. Current status

Numerous standalone software codes for optimisation are available for use by the water resources community (e.g. Eldred et al., 2006; Izzo, 2012; Matott, 2005; Pohlheim, 2007; Vesselinov and Harp, 2012). Various implementation approaches have been taken, including: binaries written in C/C++ or FORTRAN (e.g. Doherty, 2008; Duan et al., 1993; Hadka and Reed, 2013; Matott, 2005; Tolson and Shoemaker, 2007; Yan and Minsker, 2006, 2011); shared-object libraries with well-defined APIs (e.g. Keijzer et al., 2001; Levine et al., 1999; Wall, 1996); portable java code and jar files (e.g. Cingolani, 2009; Hadka, 2012; Meffert, 2012; Singh et al., 2013; White, 2012); toolboxes and packages written for python, R, perl, MatLab/Octave, and other scripted languages (e.g. Izzo, 2012; Kelley, 1999; Matott et al., 2011; Merelo Guervos et al., 2010; Regis and Shoemaker, 2005; Theussl, 2013; Vrugt et al., 2003; Vrugt and Robinson, 2007; Vrugt et al., 2008; Zlatanov, 2001); and macros and extensions to spreadsheet packages (e.g. LibreOffice, 2009; Palisade, 2013). Finally, there have been several efforts to seamlessly integrate optimisation algorithms and other “system-level” tools into comprehensive integrated modelling frameworks (e.g. Babendreier, 2003; Banta et al., 2008; David et al., 2013; Laniak et al., 2013; Leavesley et al., 2002). Overall, the community of users and developers of EAs is large and fragmented, and algorithm selection and implementation continues to be rather ad-hoc and subjective. Nonetheless, there have been some efforts to catalogue available software codes (Matott et al., 2009) and standardize search algorithm implementations (e.g. Bratton and Kennedy, 2007).

Most EA codes are intended to be general purpose and applicable to a wide variety of optimisation problems. Apart from various tuning parameters (e.g. mutation rate), these optimisers can be used in an “off-the-shelf” manner and users are shielded from the inner-workings of a given implementation. However, particular characteristics of complex real-world water resources problems may warrant ad-hoc modification of a given search algorithm (see Section 3.3). As discussed in Section 3.8, uncertainty and variability are crucial components of water resources problems, but such concerns are not typically addressed by off-the-shelf implementations. Furthermore, resolution of water resources problems typically occurs within a complex socioeconomic and legislative decision-making context (see Section 3.7). The constraints associated with such concerns can be difficult to express in a mathematical optimisation framework (see Section 3.2). Along similar lines, an abundance of soft knowledge and expert judgement is available for a given water resources problem, but such information is not easily accommodated by off-the-shelf optimisers.

EAs have usually been assessed using computationally cheap test functions during initial algorithm development and

implementation. A single objective function evaluation might take a millisecond or less. In contrast, realistic water resources problems can require hours or even days to complete a single simulation run. Recent research has sought to address this disconnect using various tactics for avoiding expensive function evaluations (see Sections 3.4 and 3.5). Implementation strategies for incorporating a given tactic into a particular algorithm are quite varied and have been largely developed in an ad-hoc fashion.

3.9.2. Research challenges and future directions

3.9.2.1. Comparing alternative codes and implementation approaches. The plethora of approaches that have been taken for algorithm implementation raises several research challenges. Firstly, it further complicates algorithm comparisons (see Section 2.4) since it is not clear that two different implementations of the same algorithm will yield identical results. Instead, it may be necessary for algorithm comparison statements to contain qualifiers, as in: “compared to the [name of comparison algorithm], as implemented in [name of package]”. There may be significant variability in performance across different implementations of the same algorithm. Yet, such possibility has not been systematically explored. Overall, the community would benefit from a comprehensive review of available codes and implementation approaches for simulation-based optimisation. A similar kind of review was recently performed for software codes related to environmental modelling (e.g. solver packages, mesh generation, etc.) (Miller et al., 2013).

3.9.2.2. Developing specialised but modular algorithms. As mentioned previously, the peculiarities of realistic water resources problems may necessitate a variety of specialised algorithm implementations. However, such specialisations should not come at the expense of good software engineering practices, like modular design and code reuse. With this in mind, a plugin approach, like the one used for customising web-browsers and other heterogeneous software packages (Miceli et al., 2013; SchedMD, 2004), may merit consideration. The basic idea is for the software to use a standard algorithm implementation if no plugin is provided. Developers could then extend algorithm behaviour by incorporating one or more plugins.

3.9.2.3. Optimisation as a service. Another recent trend is toward implementing models as services (“Model as a Service,” or MaaS), an extension of Software as a Service (SaaS, Roman et al. (2009)). MaaS enables computers to launch, analyse, and visualize model results through Web services using data standards. Optimisation as a Service (OaaS) is an obvious extension of this concept that Zhao (in preparation) is only beginning to explore. The advantage of service-oriented architectures is that multiple optimisation and simulation models can be linked in workflows and accessed through user-friendly Web applications that enable easier use and sharing. Further research is needed to explore whether and how OaaS and MaaS approaches can address the implementation issues mentioned previously.

3.9.2.4. Leveraging tools for code distribution and version control. Tools to maintain and host algorithm implementations would help mitigate the fragmentation that has occurred across users, groups, and problem types. For example, the community could do a much better job in leveraging version control platforms like Mercurial, CVS, SVN and GIT, and source code hosting services like sourceforge, github and BitBucket. Utilizing these types of platforms would help to unify the user community by ensuring that (1) all users have access to the latest software version and (2) users can contribute bug fixes, improvements, and even plugins.

3.9.2.5. Black-box vs. integrated approaches. There are two somewhat opposing approaches to simulation-based optimisation which can potentially have a significant impact on design and implementation of associated software. The “black-box” view emphasises tools that create an efficient and robust interface with existing simulation codes. In other words, the optimisation tools should be independent of the simulation model, and as such, these two should be capable of remaining wholly separate from one another. Even in the context of large-scale parallelisation, etc., this emphasis on an efficient interface has the potential to greatly facilitate general usage of optimisation tools. Alternatively, the “integrated” view argues for more explicit linkage between optimiser and simulator in the interest of maximising performance (Kannan and Wild, 2012). A plugin type of approach may allow for tighter integration with a given simulator, while still encouraging widespread tool usage.

4. Summary and conclusions

Over the last 20 years, the application of EAs to water resources problems has been an extremely active research field. Much of the research effort has been directed towards:

1. Answering the question of whether algorithms inspired by different natural phenomena or heuristics can be used successfully for solving water resources problems.
2. The development of improved optimisation algorithms, including investigation of the effect of different search operators and parameterisations.

Although EAs have exhibited potential, there is a need to shift research efforts from the development and performance assessment of EAs for relatively simple, isolated case studies, to addressing the challenges that enable them to be applied to real-world problems with efficiency and confidence in order to fully realise this potential.

A “roadmap” of the main research challenges that need to be addressed in order to move the research field of the application of EAs and other metaheuristics to water resources problems forward over the next decade is given in Fig. 4 and summarised below. In Fig. 4, the 13 research challenges are shown in the bolded, oval shapes, which are mapped into the EA process flowchart used previously as a guidemap for Sections 2 (Fig. 2) and 3 (Fig. 3). The major themes that should attract the attention of researchers in the coming years include (i) problem formulation and decision-making (Section 4.1), (ii) algorithmic issues (Section 4.2) and (iii) benchmarking (Section 4.3). It should be noted that while progress has been made in some of these areas, they are generally still in their infancy and there is a need to increasingly direct research efforts into these domains.

4.1. Problem formulation and decision-making

4.1.1. Problem formulation

Research that closely examines optimisation problem formulations for complex case studies will require increased attention in the future, including:

1. Improving the way optimisation problems are formulated so as to adequately represent the complexity of real problems (Challenge 1, Fig. 4). For example, what is the best way to (i) represent uncertainties in the formulation of the problem and to (ii) develop mathematical representations of complex objectives and constraints (e.g. environmental impact, sustainability, risk of water shortage, resilience, etc.)? In addition, it is important to

understand whether generic approaches can be developed or whether approaches are case-study specific.

2. Understanding the impacts of simplifications and assumptions in the way real problems are represented and simulated mathematically (**Challenge 2, Fig. 4**). By definition, all optimisation and simulation models are simplified representations of reality. The challenge is to quantify the impact different levels of simplification have on the optimal solutions obtained in an attempt to identify the most appropriate level of complexity for a particular problem.
3. Improving the characterisation of different sources of uncertainty, particularly in the face of deep uncertainties and simulation model structural uncertainties (**Challenge 3, Fig. 4**).

4.1.2. Decision-making

Even if the outputs from optimisation studies are reflective of the optimal solutions of the problem being addressed, the degree of benefit they can deliver is dependent on how effectively they can feed into associated decision-making processes. In particular, there is a need to:

1. Improve decision support, visualisation, and communication tools, especially when dealing with many-objective problems and related uncertainties (**Challenge 4, Fig. 4**).
2. Develop a deeper understanding of the “sociology” associated with the application of EAs to complex problems (see Section 3.2.2) in order to identify potential barriers to their

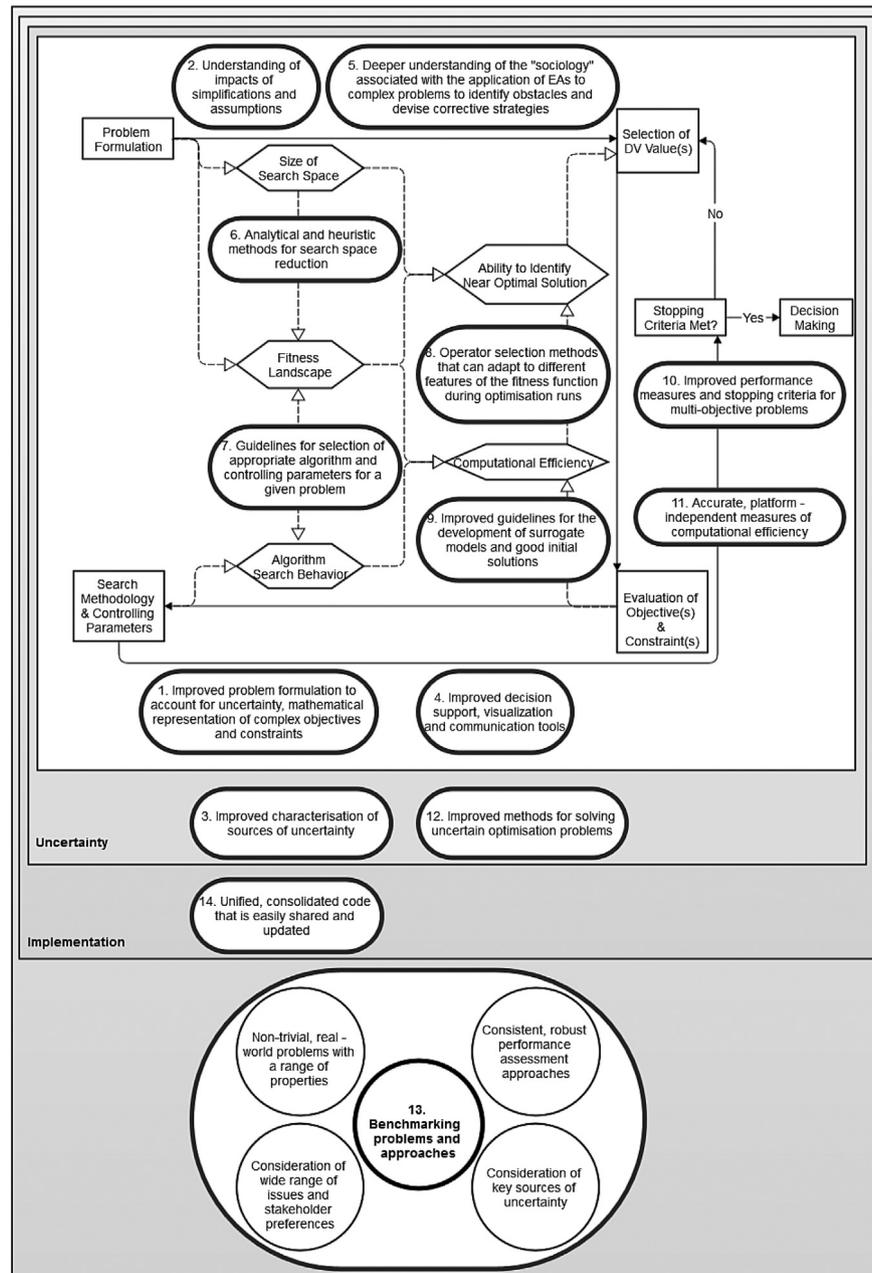


Fig. 4. Roadmap of future research challenges and directions for the application of evolutionary algorithms in water resources. The square shapes represent the steps in the EA process and the hexagonal shapes represent problem and algorithm characteristics resulting from choices made at different steps of the EA process, whereas the oval shapes represent the associated research challenges.

implementation and strategies for overcoming these barriers (**Challenge 5, Fig. 4**).

4.2. Algorithmic issues

Even though much research effort has been expended on algorithmic issues over that last two decades, the application of EAs to real-life problems creates a suite of new challenges that needs to be addressed in order to ensure that the success of the application of EAs to relatively simple problems can be translated to their successful application to real-life problems. In order to achieve this, three major research themes need to be addressed, including (i) the ability to find near-optimal solutions in a reasonable timeframe, (ii) the ability to assess algorithm performance in a robust and unbiased fashion and (iii) the ability to deal with uncertainty.

4.2.1. Ability to find near-optimal solutions in a reasonable timeframe

The ability to find (near)-optimal solutions to real water resources problems in a reasonable timeframe is a function of (i) the characteristics of the problem to be solved, (ii) the searching behaviour of the optimisation algorithm and (iii) computational efficiency, as discussed below:

4.2.1.1. Problem characteristics. The characteristics of water resources optimisation problems include (i) the size of the search space and (ii) the “ruggedness” of the fitness landscape (**Fig. 4**). In general, if the search space is larger and more “rugged”, it is more difficult to identify near-optimal solutions and it takes longer to do so. Both of these factors are a function of how the optimisation problem is defined, although algorithm searching behaviour also has an important role to play (**Fig. 4**). This further highlights the importance of the research challenges associated with problem formulation (**Challenges 1 and 2, Fig. 4**), not only in order to obtain a realistic mathematical representation of the problem to be solved, as discussed in Section 4.1, but also in order to keep the size and ruggedness of the search space to a minimum. In addition, there is a need to explore ways in which the size of the search space can be reduced, either with the aid of analytical techniques or heuristic methods developed based on domain knowledge and experience (**Challenge 6, Fig. 4**).

4.2.1.2. Searching behaviour. To enable algorithms to navigate through the search space of a particular water resources management problem effectively and efficiently, and therefore find near-optimal solutions in a reasonable timeframe, searching behaviour needs to be tailored to the characteristics of the problem being solved, as represented by the properties of the fitness function. Consequently, there is a need to:

1. Develop methods that facilitate a better understanding of (i) algorithm behaviour and (ii) the properties of the fitness landscape, which can be used develop guidelines for the selection of appropriate algorithms and the parameters that control their searching behaviour for problems with particular fitness landscape characteristics (**Challenge 7, Fig. 4**).
2. Develop searching operator selection methods that can adapt to different features of the fitness function during optimisation runs (**Challenge 8, Fig. 4**).

4.2.1.3. Computational efficiency. In the context of solving real water resources problems, computational efficiency not only has an impact on the *speed* with which near-optimal solutions are found, but also the *quality* of the solutions (including the ability to identify good solutions at all), as a result of limited computational resources

and long run-times. Overall computational efficiency is a function of the properties of the fitness landscape (i.e. the problem characteristics and algorithm searching behaviour, as discussed above), the available computational resources, the initialisation of the optimisation algorithm (i.e. the quality of the initial solutions) and the computational efficiency of the simulation model(s) used to evaluate objectives and constraints. Consequently, in addition to **Challenges 1 to 8** outlined above, there is a need to develop improved (i) methods for identifying good initial solutions and (ii) guidelines for the development of surrogate models (**Challenge 9, Fig. 4**).

4.2.2. Ability to assess algorithm performance in a robust and unbiased fashion

In order to enable the utility of the different approaches to improving an algorithm’s ability to find near-optimal solutions discussed in Section 4.2.1 to be determined, there is a need to:

1. Develop improved performance measures and stopping criteria, particularly for multi/many-objective problems (**Challenge 10, Fig. 4**).
2. Develop accurate, platform-independent measures of computational efficiency (**Challenge 11, Fig. 4**).

4.2.3. Ability to deal with uncertainty

As discussed in Section 4.1.1, adequate consideration of uncertainty in formulating water resources problems is vital when dealing with real problems. In general, this uncertainty can stem from the input data that drive the system, the calculation of objective function values, and the values for the decision variables. However, at present, there is a lack of methods for dealing with some important aspects of uncertainty. Moreover, methods such as Monte Carlo techniques can cause large increases in the computational demands in solving problems. Consequently, there is a need to develop improved methods for solving uncertain optimisation problems (**Challenge 12, Fig. 4**), including performing the calculations in a parsimonious manner.

4.3. Benchmarking

In order to assess the effectiveness of advancements in water resources problem formulation, algorithmic issues and decision making techniques in a rigorous and unbiased fashion, there is a need to develop a set of benchmarking problems and approaches, as well as consolidated, unified code for implementing them (**Challenges 13 and 14, Fig. 4**). As shown in **Fig. 4**, such benchmarking problems should (i) be non-trivial, real-world problems representative of the range of decision problems encountered in water resource planning and management, (ii) consider a wide range of issues and stakeholder preferences, (iii) consider key sources of uncertainty and (iv) be evaluated using consistent, robust performance assessment approaches (see Section 4.2.2). Application of such benchmarking problems and approaches is vital in order to provide much needed understanding and guidance on the relative utility of different optimisation methods under different circumstances, potentially leading to the development of guidelines for users. This is likely to go a long way towards enabling the full potential of EAs for solving real water resources problems to be realised.

4.4. A vision for the state-of-practice in 2025

We conclude this position paper by envisioning the state-of-practice in 10 years or so after the various challenges outlined in

this paper have been addressed. This is done both from the perspective of research and real-world applications.

4.4.1. Research

By 2025, we expect a suite of robust benchmark and case-study problems addressing realistic water resources management issues to be readily available to the EA research community via web-accessible repositories. The latest EAs and algorithm enhancements will also be readily available and served up for continued development by the community via open-source software development mechanisms.

A water resources meta-heuristics portal will emerge for curating these disparate algorithms and benchmarking efforts, serving as a focal point for ongoing dissemination of research activity. The ability for researchers to easily access algorithms and benchmark problems will lead to robust and wide-ranging algorithm analysis and comparison studies. These studies will result in the realization that just a handful of EA operators (e.g. mutation, limited memory, etc.) are responsible for optimally controlling explorative and exploitative behaviour. These studies will also result in the realization that, despite being inspired by different natural phenomena, a number of published algorithms are essentially the same and yield statistically indistinguishable results.

Furthermore, we expect a number of new algorithm implementations to embrace a plugin and/or optimization-as-a-service (OaaS) approach. These implementations will facilitate a deeper exploration of a variety of methods for handling uncertainty, ranging from conventional reliability methods to emerging multi-criteria calibration and uncertainty analysis approaches. Plugin and OaaS algorithms will also allow researchers to easily leverage available parallel computing resources as well as the latest surrogate modelling techniques.

4.4.2. Real-world applications

We expect that the challenges posed by growing population demands, degrading water quality, constrained sources of water supply, and an increasing proportion of the world's population living in mega-cities will provide significant opportunities for the application of EAs to real-world problems. By 2025, we expect to have had sufficient success in solving a range of high impact real world problems to see considerable industry engagement and uptake of EAs. Consequently, we expect that real-world decisions will be regularly influenced by the use of these optimisation approaches, and that stakeholders will begin to learn more about their systems (i.e. complicated interactions between subsystems), aided by what is learnt in optimisation. In turn, industry feedback is likely to offer even more challenging problems (particularly in the uncertainty domain), putting greater emphasis on the "sociology" of optimisation (see Section 3.2.2).

We also expect that the realisation of the above vision will be assisted by the dramatic expansion of computing capability, which has served as a major driver in the broad array of EA-based water resources applications that have emerged. Moreover, this trend is not slowing and is in fact accelerating to consider broader forms of computing and information processing that are emerging with each passing year. The water resources research community must be careful to avoid constraining their tools and thinking to current or past computing architectures. Active cloud services, ubiquitous personal computing, and new touch-based human-computer interaction are already realities. The true question is how these current technologies, as well as future expansions of computational capability, can be best brought to bear for innovative design and management of water resources systems. Consequently, from a computational perspective, the coming decade is likely to

exponentially transform the scale of water resources and decision support processes that EAs can address.

Finally, we expect that the realisation of the above vision will be supported by an expanded educational effort to train the next generation of planners who are capable of understanding how to exploit the growing EA capabilities in their problem formulations, while also having the tools necessary to minimise the unintended consequences of severe planning uncertainties, such as interactive and immersive visualisation of the complex spatial-and-temporal dynamics of evolving elements of designs. In the longer term, the successful application of EAs to real-world problems will also encourage the widespread inclusion of EA optimisation in undergraduate and postgraduate curricula.

Acknowledgements

The authors would like to thank Gayani Fernando for her assistance with putting together the references for this paper, as well as the reviewers of this paper, whose comments and suggestions have improved the quality of this paper significantly.

Appendix A. Supplementary data

Supplementary data related to this article can be found [online](#).

References

- Afshar, M., 2006. Improving the efficiency of ant algorithms using adaptive refinement: application to storm water network design. *Adv. Water Resour.* 29 (9), 1371–1382.
- Afshar, M., 2012. Rebirthing genetic algorithm for storm sewer network design. *Sci. Iran.* 19 (1), 11–19.
- Aumann, C.A., 2011. Constructing model credibility in the context of policy appraisal. *Environ. Model. Softw.* 26 (3), 258–265.
- Babbar-Sebens, M., Minsker, B., 2008. Standard interactive genetic algorithm—comprehensive optimization framework for groundwater monitoring design. *J. Water Resour. Plan. Manag.* 134 (6), 538–547.
- Babbar-Sebens, M., Minsker, B., 2010. A case-based micro interactive genetic algorithm (CBMIGA) for interactive learning and search: methodology and application to groundwater monitoring design. *Environ. Model. Softw.* 25 (10), 1176–1187. <http://dx.doi.org/10.1016/j.envsoft.2010.03.027>.
- Babbar-Sebens, M., Minsker, B., 2012. Interactive genetic algorithm with mixed initiative interaction for multi-criteria ground water monitoring design. *Appl. Soft Comput.* 12 (1), 182–195.
- Babbar-Sebens, M., Mukhopadhyay, S., 2009. Reinforcement learning for human-machine collaborative optimization: application in ground water monitoring. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC)*. IEEE, pp. 3563–3568.
- Babbar, M., Minsker, B.S., 2006. Groundwater remediation design using multiscale genetic algorithms. *J. Water Resour. Plan. Manag.* 132 (5), 341–350.
- Babendreier, J., 2003. Calibration, optimization, and sensitivity and uncertainty algorithms application programming interface (COSU-API). In: Nicholson, T.J. (Ed.), *Proceedings of the International Workshop on Uncertainty, Sensitivity, and Parameter Estimation for Multimedia Environmental Modeling*, pp. A1–A54.
- Back, T., Fogel, D.B., Michalewicz, Z., 1997. *Handbook of Evolutionary Computation*. IOP Publishing Ltd., Bristol, UK.
- Balling, R., 1999. Design by shopping: a new paradigm?. In: *Proceedings of the Third World Congress of Structural and Multidisciplinary Optimization (WCSMO-3)*, Buffalo, NY, pp. 295–297.
- Banta, E.R., Hill, M.C., Poeter, E., Doherty, J.E., Babendreier, J., 2008. Building model analysis applications with the Joint Universal Parameter Identification and Evaluation of Reliability (JUPITER) API. *Comput. Geosci.* 34 (4), 310–319. <http://dx.doi.org/10.1016/j.cageo.2007.03.016>.
- Basdekas, L., 2014. Is multiobjective optimization ready for water resources practitioners? Utility's drought policy investigation. *J. Water Resour. Plan. Manag.* 140 (3), 275–276.
- Basupi, I., Kapelan, Z., Flexible water distribution system design under future demand uncertainty. *J. Water Resour. Plan. Manag.* ASCE. [http://dx.doi.org/10.1061/\(ASCE\)WR.1943-5452.0000416](http://dx.doi.org/10.1061/(ASCE)WR.1943-5452.0000416).
- Bayer, P., de Paly, M., Buerger, C.M., 2010. Optimization of high-reliability-based hydrological design problems by robust automatic sampling of critical model realizations. *Water Resour. Res.* 46 <http://dx.doi.org/10.1029/2009wr008081>.

- Bayer, P., Finkel, M., 2004. Evolutionary algorithms for the optimization of advective control of contaminated aquifer zones. *Water Resour. Res.* 40, W06506. <http://dx.doi.org/10.1029/2003WR002675>.
- Bayer, P., Finkel, M., 2007. Optimization of concentration control by evolution strategies: formulation, application, and assessment of remedial solutions. *Water Resour. Res.* 43 (2), W02410. <http://dx.doi.org/10.1029/2005WR004753>.
- Becker, D., Minsker, B., Greenwald, R., Zhang, Y., Harre, K., Yager, K., Zheng, C., Peralta, R., 2006. Reducing long term remedial costs by transport modeling optimization. *Ground Water* 44 (6), 864–875.
- Beh, E.H., Dandy, G.C., Maier, H.R., Paton, F.L., 2014. Optimal sequencing of water supply options at the regional scale incorporating alternative water supply sources and multiple objectives. *Environ. Model. Softw.* 53, 137–153. <http://dx.doi.org/10.1016/j.envsoft.2013.11.004>.
- Beh, E.H.Y., Maier, H.R., Dandy, G.C., 2011. Development of a modelling framework for optimal sequencing of water supply options at the regional scale incorporating sustainability and uncertainty. In: Chan, F., Marinova, D., Anderssen, R.S. (Eds.), *Proceedings of the 19th International Congress on Modelling and Simulation*, Perth, Australia.
- Behzadian, K., Kapelan, Z., Savic, D., Ardeshir, A., 2009. Stochastic sampling design using a multi-objective genetic algorithm and adaptive neural networks. *Environ. Model. Softw.* 24 (4), 530–541. <http://dx.doi.org/10.1016/j.envsoft.2008.09.013>.
- Bellman, R., 1961. On the approximation of curves by line segments using dynamic programming. *Commun. ACM* 4 (6), 284.
- Beven, K., 2006. A manifesto for the equifinality thesis. *J. Hydrol.* 320 (1), 18–36.
- Bhandari, D., Murrthy, C.A., Pal, S.K., 2012. Variance as a stopping criterion for genetic algorithms with elitist model. *Fundam. Informa.* 120 (2), 145–164. <http://dx.doi.org/10.3233/fi-2012-754>.
- Bi, W., Dandy, G.C., 2013. Optimization of water distribution systems using online retrained metamodels. *J. Water Resour. Plan. Manag.* [http://dx.doi.org/10.1061/\(ASCE\)WR.1943-5452.0000419,04014032](http://dx.doi.org/10.1061/(ASCE)WR.1943-5452.0000419,04014032).
- Bieker, H.P., Slupphaug, O., Johansen, T.A., 2007. Real-time production optimization of oil and gas production systems: a technology survey. *Spe Prod. Oper.* 22 (4), 382–391.
- Birattari, M., Zloch, M., Dorigo, M., 2006. Towards a theory of practice in metaheuristics design: a machine learning perspective. *RAIRO Theor. Inform. Appl.* 40 (02), 353–369.
- Blanning, R.W., 1975. Construction and implementation of metamodels. *Simulation* 24 (6), 177–184. <http://dx.doi.org/10.1177/003754977502400606>.
- Blum, C., Roli, A., 2003. Metaheuristics in combinatorial optimization: overview and conceptual comparison. *ACM Comput. Surv.* 35 (3), 268–308.
- Boese, K.D., Kahng, A.B., Muddu, S., 1994. A new adaptive multi-start technique for combinatorial global optimizations. *Oper. Res. Lett.* 16 (2), 101–113.
- Bowden, G.J., Dandy, G.C., Maier, H.R., 2005a. Input determination for neural network models in water resources applications. Part 1—background and methodology. *J. Hydrol.* 301 (1), 75–92.
- Bowden, G.J., Maier, H.R., Dandy, G.C., 2005b. Input determination for neural network models in water resources applications. Part 2. Case study: forecasting salinity in a river. *J. Hydrol.* 301 (1), 93–107.
- Branke, J., Deb, K., Miettinen, K., Slowinski, R. (Eds.), 2008. *Multiobjective Optimization: Interactive and Evolutionary Approaches*. Lecture Notes in Computer Science, vol. 5252. Springer, Berlin-Heidelberg, D. (ch 8 and 9).
- Bratton, D., Kennedy, J., 2007. Defining a standard for particle swarm optimization. In: *IEEE Swarm Intelligence Symposium*, pp. 120–127. <http://dx.doi.org/10.1109/sis.2007.368035>.
- Brill Jr., E.D., Flach, J.M., Hopkins, L.D., Ranjithan, S., 1990. MGA: a decision support system for complex, incompletely defined problems. *IEEE Trans. Syst. Man Cybern.* 20 (4), 745–757.
- Broad, D.R., Dandy, G.C., Maier, H.R., 2005. Water distribution system optimization using metamodels. *J. Water Resour. Plan. Manag.* ASCE 131 (3), 172–180. [http://dx.doi.org/10.1061/\(asce\)0733-9496\(2005\)131:3\(172\)](http://dx.doi.org/10.1061/(asce)0733-9496(2005)131:3(172)).
- Broad, D.R., Maier, H.R., Dandy, G.C., 2010. Optimal operation of complex water distribution systems using metamodels. *J. Water Resour. Plan. Manag.* ASCE 136 (4), 433–443. [http://dx.doi.org/10.1061/\(asce\)wr.1943-5452.0000052](http://dx.doi.org/10.1061/(asce)wr.1943-5452.0000052).
- Brockhoff, D., Zitzler, E., 2009. Objective reduction in evolutionary multiobjective optimization: theory and applications. *Evol. Comput.* 17 (2), 135–166.
- Cantu-Paz, E., 2000. *Efficient and Accurate Parallel Genetic Algorithms*. Kluwer.
- Castelletti, A., Antenucci, J., Limosani, D., Quach Thi, X., Soncini-Sessa, R., 2011. Interactive response surface approaches using computationally intensive models for multiobjective planning of lake water quality remediation. *Water Resour. Res.* 47 (9).
- Castelletti, A., Galelli, S., Ratto, M., Soncini-Sessa, R., Young, P.C., 2012a. A general framework for Dynamic Emulation Modelling in environmental problems. *Environ. Model. Softw.* 34, 5–18. <http://dx.doi.org/10.1016/j.envsoft.2012.01.002>.
- Castelletti, A., Galelli, S., Restelli, M., Soncini-Sessa, R., 2012b. Data-driven dynamic emulation modelling for the optimal management of environmental systems. *Environ. Model. Softw.* 34, 30–43.
- Castelletti, A., Lotov, A., Soncini-Sessa, R., 2010a. Visualization-based multi-objective improvement of environmental decision-making using linearization of response surfaces. *Environ. Model. Softw.* 25 (12), 1552–1564.
- Castelletti, A., Pianosi, F., Soncini-Sessa, R., Antenucci, J.P., 2010b. A multiobjective response surface approach for improved water quality planning in lakes and reservoirs. *Water Resour. Res.* 46 (6), W06502. <http://dx.doi.org/10.1029/2009WR008389>.
- Castelletti, A., Soncini-Sessa, R., 2006. A procedural approach to strengthening integration and participation in water resource planning. *Environ. Model. Softw.* 21 (10), 1455–1470. <http://dx.doi.org/10.1016/j.envsoft.2005.07.013>.
- Castelletti, A., Yajima, H., Giuliani, M., Soncini-Sessa, R., Weber, E., 2014. Planning the optimal operation of a multioutlet water reservoir with water quality and quantity targets. *J. Water Resour. Plan. Manag.* 140 (4), 496–510. [http://dx.doi.org/10.1061/\(ASCE\)WR.1943-5452.0000348](http://dx.doi.org/10.1061/(ASCE)WR.1943-5452.0000348).
- Cheung, P.B., Reis, L.F., Formiga, K.T., Chaudhry, F.H., Ticona, W.G., 2003. Multi-objective evolutionary algorithms applied to the rehabilitation of a water distribution system: a comparative study. In: *Evolutionary Multi-criterion Optimization*. Lecture Notes in Computer Science 2632. Springer Verlag, pp. 662–676.
- Cingolani, P., 2009. *Jswarm-pso - an Open Source Swarm Optimization Package*. Available from: <http://jswarm-pso.sourceforge.net/>.
- Coello, C.C., Lamont, G.B., Van Veldhuizen, D.A., 2007. *Evolutionary Algorithms for Solving Multi-objective Problems*, second ed. Springer, New York.
- Collette, Y., Siarry, P., 2003. *Multiobjective Optimization: Principles and Case Studies*. Springer.
- Collette, Y., Siarry, P., 2005. Three new metrics to measure the convergence of metaheuristics towards the Pareto frontier and the aesthetic of a set of solutions in biobjective optimization. *Comput. Oper. Res.* 32 (4), 773–792.
- Colorni, A., Dorigo, M., Maffioli, F., Maniezzo, V., Righini, G., Trubian, M., 1996. Heuristics from nature for hard combinatorial optimization problems. *Int. Trans. Oper. Res.* 3 (1), 1–21.
- Cui, L., Kuczera, G., 2005. Optimizing water supply headworks operating rules under stochastic inputs: assessment of genetic algorithm performance. *Water Resour. Res.* 41 (5), W05016. <http://dx.doi.org/10.1029/2004wr003517>.
- Culberson, J., Lichtner, J., 1996. On searching 'alpha' - ary hypercubes and related graphs. In: *Belew, R., Vose, M. (Eds.), Foundations of Genetic Algorithms*. Morgan Kaufmann, pp. 263–290.
- Cunha, M.C., Ribeiro, L., 2004. Tabu search algorithms for water network optimization. *Eur. J. Oper. Res.* 157 (3), 746–758.
- da Fonseca, V.G., Fonseca, C.M., Hall, A.O., 2001. Inferential performance assessment of stochastic optimisers and the attainment function. In: *Proceedings of the Evolutionary Multi-criterion Optimization*. Springer, pp. 213–225.
- David, O., Ascough II, J.C., Lloyd, W., Green, T.R., Rojas, K.W., Leavesley, G.H., Ahuja, L.R., 2013. A software engineering perspective on environmental modeling framework design: the object modeling system. *Environ. Model. Softw.* 39, 201–213. <http://dx.doi.org/10.1016/j.envsoft.2012.03.006>.
- Davidov, Y., 1991. Epistasis variance: a viewpoint on GA-hardness. In: *Rawlins, G.J.E. (Ed.), Foundations of Genetic Algorithms*. Morgan Kaufmann, pp. 23–35.
- De Jong, K., Spears, W., Gordan, D., 1995. Using markov chains to analyse GAFOs. In: *Whitley, L.D., Vose, M.D. (Eds.), Foundations of Genetic Algorithms 3*. Morgan Kaufmann, pp. 115–137.
- Deb, K., 2001. *Multi-objective Optimization Using Evolutionary Algorithms*. John Wiley, NY.
- Deb, K., Goldberg, D.E., 1994. Sufficient conditions for deceptive and easy binary functions. *Ann. Math. Artif. Intell.* 10 (4), 385–408.
- Deb, K., Gupta, H., 2006. Introducing robustness in multi-objective optimization. *Evol. Comput.* 14 (4), 463–494. <http://dx.doi.org/10.1162/evco.2006.14.4.463>.
- Deb, K., Jain, S., 2002. *Running Performance Metrics for Evolutionary Multiobjective Optimization*. Indian Institute of Technology Kanpur.
- Deb, K., Sundar, J., Udaya Bhaskara Rao, N., Chaudhuri, S., 2006. Reference point based multi-objective optimization using evolutionary algorithms. *Int. J. Comput. Intell. Res.* 2 (3), 273–286.
- di Pierro, F., Khu, S.-T., Savic, D.A., 2007. An investigation on preference order ranking scheme for multiobjective evolutionary optimization. *Evol. Comput. IEEE Trans.* 11 (1), 17–45.
- Dobes, L., 2008. Getting real about adapting to climate change: using 'real options' to address the uncertainties. *Agenda J. Policy Anal. Reform* 15 (3), 55–69.
- Doherty, J., 2008. Addendum to the PEST Manual. Watermark Numerical Computing, Brisbane, Australia.
- Dougherty, D.E., Marryott, R.A., 1991. Optimal groundwater management: 1. Simulated annealing. *Water Resour. Res.* 27 (10), 2493–2508.
- Droste, S., Jansen, T., Wegener, I., 1999. Perhaps not a free lunch but at least a free appetizer. In: *GECCO*. Morgan Kaufmann, pp. 833–839.
- Duan, Q.Y., Gupta, V.K., Sorooshian, S., 1993. A shuffled complex evolution approach for effective and efficient global minimization. *J. Optim. Theory Appl.* 76 (3), 501–521. <http://dx.doi.org/10.1007/bf00939380>.
- Eldred, M.S., Brown, S.L., Adams, B.M., Dunlavy, D.M., Gay, D.M., Swiler, L.P., Giunta, A.A., Hart, W.E., Watson, J.P., Eddy, J.P., Griffin, J.D., Hough, P.D., Kolda, T.G., Martinez-Canales, M.L., Williams, P.J., 2006. *DAKOTA, a Multilevel Parallel Object-oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 4.0 Users Manual*. Sandia National Laboratories.
- Fernando, T.M.K.G., Maier, H.R., Dandy, G.C., 2009. Selection of input variables for data driven models: an average shifted histogram partial mutual information estimator approach. *J. Hydrol.* 367 (3), 165–176.
- Fleming, P.J., Purshouse, R.C., Lygoe, R.J., 2005. Many-objective optimization: an engineering design perspective. *Lecture Notes in Computer Science 3410: Evolutionary Multi-criterion Optimization*. Springer, Berlin, pp. 14–32.
- Foong, W.K., Maier, H.R., Simpson, A.R., 2008a. Power plant maintenance scheduling using ant colony optimization: an improved formulation. *Eng. Optim.* 40 (4), 309–329. <http://dx.doi.org/10.1080/03052150701775953>.

- Foong, W.K., Simpson, A.R., Maier, H.R., Stolp, S., 2008b. Ant colony optimization for power plant maintenance scheduling optimization—a five-station hydropower system. *Ann. Oper. Res.* 159 (1), 433–450. <http://dx.doi.org/10.1007/s10479-007-0277-y>.
- Franssen, M., 2005. Arrow's theorem, multi-criteria decision problems and multi-attribute preferences in engineering design. *Res. Eng. Des.* 16 (1–2), 42–56.
- Fu, G., Kapelan, Z., 2011. Fuzzy probabilistic design of water distribution networks. *Water Resour. Res.* 47, W05538. <http://dx.doi.org/10.1029/2010wr009739>.
- Fu, G., Kapelan, Z., Kasprzyk, J.R., Reed, P., 2012. Optimal design of water distribution systems using many-objective visual analytics. *J. Water Resour. Plan. Manag.* 139 (6), 624–633.
- Fu, G., Kapelan, Z., Reed, P., 2011. Reducing the complexity of multiobjective water distribution system optimization through global sensitivity analysis. *J. Water Resour. Plan. Manag. ASCE* 138 (3), 196–207.
- Galelli, S., Castelletti, A., 2013. Tree-based iterative input variable selection for hydrological modeling. *Water Resour. Res.* 49 (7), 4295–4310. <http://dx.doi.org/10.1002/wrcr.20339>.
- Galelli, S., Humphrey, G.B., Maier, H.R., Castelletti, A., Dandy, G.C., Gibbs, M.S., December 2014. An evaluation framework for input variable selection algorithms for environmental data-driven models. *Environ. Model. Softw.* 62, 33–51. <http://dx.doi.org/10.1016/j.envsoft.2014.08.015>.
- Gaspar-Cunha, A., Vieira, A., 2005. A multi-objective evolutionary algorithm using neural networks to approximate fitness evaluations. *Int. J. Comput. Syst. Signals* 6 (1), 18–36.
- Gibbs, M.S., Dandy, G.C., Maier, H.R., 2008. A genetic algorithm calibration method based on convergence due to genetic drift. *Inf. Sci.* 178 (14), 2857–2869. <http://dx.doi.org/10.1016/j.ins.2008.03.012>.
- Gibbs, M.S., Maier, H.R., Dandy, G.C., 2010. Comparison of genetic algorithm parameter setting methods for chlorine injection optimization. *J. Water Resour. Plan. Manag. ASCE* 136 (2), 288–291. [http://dx.doi.org/10.1061/\(asce\)wr.1943-5452.0000033](http://dx.doi.org/10.1061/(asce)wr.1943-5452.0000033).
- Gibbs, M.S., Maier, H.R., Dandy, G.C., 2011. Relationship between problem characteristics and the optimal number of genetic algorithm generations. *Eng. Optim.* 43 (4), 349–376. <http://dx.doi.org/10.1080/0305215X.2010.491547>.
- Giuliani, M., Galelli, S., Soncini-Sessa, R., 2014a. A dimensionality reduction approach for many-objective Markov Decision Processes: application to a water reservoir operation problem. *Environ. Model. Softw.* 57, 101–114.
- Giuliani, M., Herman, J.D., Castelletti, A., Reed, P., 2014b. Many-objective reservoir policy identification and refinement to reduce policy inertia and myopia in water management. *Water Resour. Res.* 50 (4), 3355–3377. <http://dx.doi.org/10.1002/2013WR014700>.
- Goh, C.-K., Tan, K.C., 2009. *Evolutionary Multi-objective Optimization in Uncertain Environments: Issues and Algorithms*. Springer.
- Goldberg, D.E., 1989. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading.
- Goldberg, D.E., 2002. *The Design of Innovation: Lessons from and for Competent Genetic Algorithms*. Kluwer Academic Publishers.
- Goldberg, D.E., Korb, B., Deb, K., 1989. Messy genetic algorithms: motivation, analysis, and first results. *Complex Syst.* 3 (5), 493–530.
- Gopalakrishnan, G., Minsker, B., Goldberg, D., 2003. Optimal sampling in a noisy genetic algorithm for risk-based remediation design. *J. Hydroinform.* 5 (1), 11–25.
- Gorissen, D., Couckuyt, I., Demeester, P., Dhaene, T., Crombecq, K., 2010. A surrogate modeling and adaptive sampling toolbox for computer based design. *J. Mach. Learn. Res.* 11, 2051–2055.
- Grayman, W.M., Loucks, D.P., Saito, L., 2012. *Toward a Sustainable Water Future: Visions for 2050*. ASCE.
- Guo, H., Hsu, W.H., 2003. GA-hardness revisited. In: Goos, G., Hartmanis, J., vanLeeuwen, J. (Eds.), *GECCO 2003, Lecture Notes in Computer Science* 2724. Springer, pp. 1584–1585.
- Guo, Y., Keedwell, E.C., Walters, G.A., Khu, S.-T., 2007. Hybridizing cellular automata principles and NSGALL for multi-objective design of urban water networks. In: *Evolutionary Multi-criterion Optimization*. Springer, pp. 546–559.
- Gupta, H.V., Sorooshian, S., Yapo, P.O., 1998. Toward improved calibration of hydrologic models: multiple and noncommensurable measures of information. *Water Resour. Res.* 34 (4), 751–763.
- Gupta, H.V., Wagener, T., Liu, Y., 2008. Reconciling theory with observations: elements of a diagnostic approach to model evaluation. *Hydrol. Process.* 22 (18), 3802–3813.
- Hadka, D., 2012. MOEA Framework: a Free and Open Source Java Framework for Multiobjective Optimization. Available from: <http://www.moeaframework.org/>.
- Hadka, D., Reed, P., 2012. Diagnostic assessment of search controls and failure modes in many-objective evolutionary optimization. *Evol. Comput.* 20 (3), 423–452.
- Hadka, D., Reed, P., 2013. Borg: an auto-adaptive many-objective evolutionary computing framework. *Evol. Comput.* 21 (2), 231–259.
- Haimes, Y., 1998. *Risk Modeling, Assessment, and Management*. John Wiley & Sons, Inc., New York.
- Haimes, Y.Y., Hall, W.A., 1977. Sensitivity, responsivity, stability and irreversibility as multiple objectives in civil systems. *Adv. Water Resour.* 1 (2), 71–81.
- Halhal, D., Walters, G.A., Ouazar, D., Savic, D., 1997. Water network rehabilitation with structured messy genetic algorithm. *J. Water Resour. Plan. Manag. ASCE* 123 (3), 137–146.
- Hiew, K.L., Labadie, J.W., Scott, J.F., 1989. Optimal operational analysis of the Colorado-Big Thompson project. In: *Computerized Decision Support Systems for Water Managers*. ASCE, pp. 632–646.
- Hsu, N.-S., Cheng, K.-W., 2002. Network flow optimization model for basin-scale water supply planning. *J. Water Resour. Plan. Manag. ASCE* 128 (2), 102–112.
- Hutton, C.J., Kapelan, Z., Vamvakieridou-Lyroudia, L., Savic, D.A., 2014. Dealing with uncertainty in water distribution system models: a framework for real-time modeling and data assimilation. *J. Water Resour. Plan. Manag. ASCE* 140 (2), 119–133. [http://dx.doi.org/10.1061/\(asce\)wr.1943-5452.0000325](http://dx.doi.org/10.1061/(asce)wr.1943-5452.0000325).
- Hwang, C.-L., Masud, A.S.M., Paidy, S.R., Yoon, K.P., 1979. Multiple objective decision making, methods and applications: a state-of-the-art survey. In: *Lecture Notes in Economics and Mathematical Systems*, vol. 164. Springer Berlin.
- Hyde, K.M., Maier, H.R., 2006. Distance-based and stochastic uncertainty analysis for multi-criteria decision analysis in Excel using Visual Basic for Applications. *Environ. Model. Softw.* 21 (12), 1695–1710.
- Hyde, K.M., Maier, H.R., Colby, C.B., 2004. Reliability-based approach to multicriteria decision analysis for water resources. *J. Water Resour. Plan. Manag.* 130 (6), 429–438.
- Hyde, K.M., Maier, H.R., Colby, C.B., 2005. A distance-based uncertainty analysis approach to multi-criteria decision analysis for water resource decision making. *J. Environ. Manag.* 77 (4), 278–290.
- Inselberg, A., 1997. *Multidimensional Detective*. In: *IEEE Symposium on Information Visualization*, pp. 100–107.
- Izzo, D., 2012. PyGMO and PyKEP: open source tools for massively parallel optimization in astrodynamics. In: *Proceedings of the 5th International Conference on Astrodynamics Tools and Techniques*. ICATT, Noordwijk, Netherlands.
- Jacobs, J., Freeman, G., Grygier, J., Morton, D., Schultz, G., Staschus, K., Stedinger, J., 1995. SOCRATES - a system for scheduling hydroelectric generation under uncertainty. *Ann. Oper. Res.* 59, 99–133. <http://dx.doi.org/10.1007/bf02031745>.
- Jakeman, A.J., Letcher, R.A., 2003. Integrated assessment and modelling: features, principles and examples for catchment management. *Environ. Model. Softw.* 18 (6), 491–501. [http://dx.doi.org/10.1016/s1364-8152\(03\)00024-0](http://dx.doi.org/10.1016/s1364-8152(03)00024-0).
- Jin, Y., 2005. A comprehensive survey of fitness approximation in evolutionary computation. *Soft Comput.* 9 (1), 3–12. <http://dx.doi.org/10.1007/s00500-003-0328-5>.
- Jones, D.F., Mirrazavi, S.K., Tamiz, M., 2002. Multi-objective meta-heuristics: an overview of the current state-of-the-art. *Eur. J. Oper. Res.* 137 (1), 1–9. [http://dx.doi.org/10.1016/s0377-2217\(01\)00123-0](http://dx.doi.org/10.1016/s0377-2217(01)00123-0).
- Jones, T., 1995. *Evolutionary Algorithms, Fitness Landscapes and Search* (PhD thesis). The University of New Mexico.
- Kadu, M.S., Gupta, R., Bhave, P.R., 2008. Optimal design of water networks using a modified genetic algorithm with reduction in search space. *J. Water Resour. Plan. Manag. ASCE* 134 (2), 147–160.
- Kallel, L., 1998. Inside GA dynamics: ground basis for comparison. In: *Parallel Problem Solving from Nature—PPSN V 1498*, pp. 57–66.
- Kallel, L., Naudts, B., Schoenauer, M., 1999. On functions with a given fitness-distance relation. In: Angeline, P.J., Michalewicz, Z., Schoenauer, M., Yao, X., Zalzal, A. (Eds.), *Proceedings of the Congress on Evolutionary Computation, CEC 99*. IEEE, Mayflower Hotel, Washington D.C., USA, pp. 1910–1916.
- Kang, D., Lansey, K., 2011. Revisiting optimal water-distribution system design: issues and a heuristic hierarchical approach. *J. Water Resour. Plan. Manag. ASCE* 138 (3), 208–217.
- Kannan, A., Wild, S.M., 2012. Benefits of deeper analysis in simulation-based groundwater optimization problems. In: *Proceedings of the XIX International Conference on Computational Methods in Water Resources (CMWR 2012)*, University of Illinois at Urbana-Champaign.
- Kapelan, Z.S., Savic, D.A., Walters, G.A., 2005. Multiobjective design of water distribution systems under uncertainty. *Water Resour. Res.* 41 (11), W11407. <http://dx.doi.org/10.1029/2004wr003787>.
- Kasprzyk, J.R., Nataraj, S., Reed, P.M., Lempert, R.J., 2013. Many objective robust decision making for complex environmental systems undergoing change. *Environ. Model. Softw.* 42, 55–71.
- Kasprzyk, J.R., Reed, P.M., Characklis, G.W., Kirsch, B.R., 2012. Many-objective de Novo water supply portfolio planning under deep uncertainty. *Environ. Model. Softw.* 34, 87–104.
- Kasprzyk, J.R., Reed, P.M., Kirsch, B.R., Characklis, G.W., 2009. Managing population and drought risks using many-objective water portfolio planning under uncertainty. *Water Resour. Res.* 45, W12401. <http://dx.doi.org/10.1029/2009wr008121>.
- Katada, Y., Ohkura, K., 2006. Estimating the degree of neutrality in fitness landscapes by the nei's standard genetic distance—an application to evolutionary robotics. In: *Proceedings of the 2006 IEEE Congress on Evolutionary Computation (CEC2006)*, pp. 483–490.
- Kavetski, D., Kuczera, G., Franks, S.W., 2006a. Bayesian analysis of input uncertainty in hydrological modeling: 1. Theory. *Water Resour. Res.* 42 (3), W03407. <http://dx.doi.org/10.1029/2005wr004368>.
- Kavetski, D., Kuczera, G., Franks, S.W., 2006b. Bayesian analysis of input uncertainty in hydrological modeling: 2. Application. *Water Resour. Res.* 42 (3), W03408. <http://dx.doi.org/10.1029/2005wr004376>.
- Keating, E.H., Doherty, J., Vrugt, J.A., Kang, Q., 2010. Optimization and uncertainty assessment of strongly nonlinear groundwater models with high parameter dimensionality. *Water Resour. Res.* 46, W10517. <http://dx.doi.org/10.1029/2009wr008584>.
- Keedwell, E., Khu, S.T., 2005. A hybrid genetic algorithm for the design of water distribution networks. *Eng. Appl. Artif. Intell.* 18 (4), 461–472.
- Keeney, R.L., Raiffa, H., 1976. *Decisions with Multiple Objectives*. John Wiley, New York.

- Keeney, R.L., Raiffa, H., 1993. *Decisions with Multiple Objectives: Preferences and Value Trade-offs*. Cambridge University Press, Cambridge, UK.
- Keijzer, M., Merelo, J.J., Romero, G., Schoenauer, M., 2001. Evolving objects: yet another evolutionary computation library. In: 5th International Conference on Artificial Evolution, Université de Bourgogne, France.
- Kelley, C.T., 1999. *Iterative Methods for Optimization*. SIAM, Philadelphia.
- Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Proceedings of IEEE International Conference on Neural Networks, Perth, Australia, pp. 1942–1948. <http://dx.doi.org/10.1109/ICNN.1995.488968>.
- Khu, S.T., Savic, D., Liu, Y., Madsen, H., 2004. A fast evolutionary based meta-modelling approach for the calibration of a rainfall-runoff model. In: Proceedings of the 1st Biennial Meeting of the Int. Environmental Modelling Software Society, Osnabruck, Germany.
- Kinnear Jr., K.E., 1994. Fitness landscapes and difficulty in genetic programming. In: Proceedings of the 1994 IEEE World Congress on Computational Intelligence. IEEE, pp. 142–147.
- Kleijnen, J.P.C., 2009. Kriging metamodeling in simulation: a review. *Eur. J. Oper. Res.* 192 (3), 707–716. <http://dx.doi.org/10.1016/j.ejor.2007.10.013>.
- Knowles, J., 2006. ParEGO: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Trans. Evol. Comput.* 10 (1), 50–66. <http://dx.doi.org/10.1109/tevc.2005.851274>.
- Knowles, J., Corne, D., 2002. On metrics for comparing nondominated sets. In: CEC'02. Proceedings of the 2002 Congress on Evolutionary Computation. IEEE, pp. 711–716.
- Knowles, J., Nakayama, H., 2008. Meta-modeling in multiobjective optimization. In: Branke, J., Deb, K., Miettinen, K., Slowinski, R. (Eds.), *Lecture Notes in Computer Science 5252, Multiobjective Optimization: Interactive and Evolutionary Approaches*. Springer, pp. 245–284.
- Kollat, J.B., Reed, P., Kasprzyk, J., 2008. A new epsilon-dominance hierarchical Bayesian optimization algorithm for large multiobjective monitoring network design problems. *Adv. Water Resour.* 31 (5), 828–845.
- Kollat, J.B., Reed, P.M., 2006. Comparing state-of-the-art evolutionary multi-objective algorithms for long-term groundwater monitoring design. *Adv. Water Resour.* 29 (6), 792–807.
- Kollat, J.B., Reed, P.M., 2007a. A computational scaling analysis of multiobjective evolutionary algorithms in long-term groundwater monitoring applications. *Adv. Water Resour.* 30 (3), 408–419.
- Kollat, J.B., Reed, P.M., 2007b. A framework for visually interactive decision-making and design using evolutionary multiobjective optimization (VIDEO). *Environ. Model. Softw.* 22 (12), 1691–1704.
- Kollat, J.B., Reed, P.M., Maxwell, R.M., 2011. Many-objective groundwater monitoring network design using bias-aware ensemble kalman filtering, evolutionary optimization, and visual analytics. *Water Resour. Res.* 47, W02529. <http://dx.doi.org/10.1029/2010WR009194>.
- Korteling, B., Dessai, S., Kapelan, Z., 2013. Using information-gap decision theory for water resources planning under severe uncertainty. *Water Resour. Manag.* 27 (4), 1149–1172. <http://dx.doi.org/10.1007/s11269-012-0164-4>.
- Kourakos, G., Mantoglou, A., 2008. Remediation of heterogeneous aquifers based on multiobjective optimization and adaptive determination of critical realizations. *Water Resour. Res.* 44 (12) <http://dx.doi.org/10.1029/2008wr007108>.
- Labadie, J.W., 2004. Optimal operation of multireservoir systems: state-of-the-art review. *J. Water Resour. Plan. Manag. ASCE* 130 (2), 93–111. [http://dx.doi.org/10.1061/\(asce\)0733-9496\(2004\)130:2\(93\)](http://dx.doi.org/10.1061/(asce)0733-9496(2004)130:2(93)).
- Laloy, E., Rogiers, B., Vrugt, J.A., Mallants, D., Jacques, D., 2013. Efficient posterior exploration of a high dimensional groundwater model from two stage Markov chain Monte Carlo simulation and polynomial chaos expansion. *Water Resour. Res.* 49 (5), 2664–2682. <http://dx.doi.org/10.1002/wrcr.20226>.
- Laniak, G.F., Olchin, G., Goodall, J., Voinov, A., Hill, M., Glynn, P., Whelan, G., Geller, G., Quinn, N., Blind, M., Peckham, S., Reaney, S., Gaber, N., Kennedy, R., Hughes, A., 2013. Integrated environmental modeling: a vision and roadmap for the future. *Environ. Model. Softw.* 39, 3–23. <http://dx.doi.org/10.1016/j.envsoft.2012.09.006>.
- Larichev, O., 1992. Cognitive validity in design of decision-aiding techniques. *J. Multi-criteria Decis. Anal.* 1 (3), 127–138.
- Laumanns, M., Thiele, L., Deb, K., Zitzler, E., 2002. Combining convergence and diversity in evolutionary multiobjective optimization. *Evol. Comput.* 10 (3), 263–282.
- Leavesley, G.H., Markstrom, S.L., Restrepo, P.J., Viger, R.J., 2002. A modular approach to addressing model design, scale, and parameter estimation issues in distributed hydrological modelling. *Hydrol. Process.* 16 (2), 173–187. <http://dx.doi.org/10.1002/hyp.344>.
- Lempert, R.J., 2002. A new decision sciences for complex systems. In: Proceedings of the National Academy of Sciences, United States of America, 99, pp. 7309–7313.
- Levine, D., Hallstrom, P., Noelle, D., Walenz, B., 1999. Experiences with the PGAPack parallel genetic algorithm library. In: Davis, L., Jong, K., Vose, M., Whitley, L.D. (Eds.), *Evolutionary Algorithms*. Springer, pp. 139–149.
- LibreOffice, 2009. Solver for Nonlinear Programming. Available from: <http://www.libreoffice.org/features/extensions>.
- Liebman, J.C., 1976. Some simple-minded observations on the role of optimization in public systems decision-making. *Interfaces* 6 (4), 102–108.
- López-Ibáñez, M., Paquete, L., Stützle, T., 2010. Exploratory analysis of stochastic local search algorithms in biobjective optimization. In: *Experimental Methods for the Analysis of Optimization Algorithms*. Springer, pp. 209–222.
- López-Ibáñez, M., Stützle, T., 2012. An experimental analysis of design choices of multi-objective ant colony optimization algorithms. *Swarm Intell.* 6 (3), 207–232.
- Lotov, A.V., Bourmistrova, L.V., Efremov, R.V., Bushenkov, V.A., Buber, A.L., Brainin, N., 2005. Experience of model integration and Pareto frontier visualization in the search for preferable water quality strategies. *Environ. Model. Softw.* 20 (2), 243–260.
- Lotov, A.V., Miettinen, K., 2008. *Visualizing the Pareto frontier*. In: *Multiobjective Optimization*. Springer, pp. 213–243.
- Loucks, D.P., 2012. Water resource management modeling in 2050. In: Grayman, W.M., Loucks, D.P., Saito, L. (Eds.), *Toward a Sustainable Water Future: Visions for 2050*. ASCE, 341.
- Madetoja, E., Ruotsalainen, H., Monkkonen, V.-M., Hamalainen, J., Deb, K., 2008. Visualizing multi-dimensional Pareto-optimal fronts with a 3D virtual reality system. In: Proceedings of the International Multiconference on Computer Science and Information Technology, Wisla, Poland, pp. 907–913.
- Malan, K.M., Engelbrecht, A.P., 2013. A survey of techniques for characterising fitness landscapes and some possible ways forward. *Inf. Sci.* 241, 148–163. <http://dx.doi.org/10.1016/j.ins.2013.04.015>.
- Marchi, A., Dandy, G.C., Wilkins, A., Rohrlach, H., 2014. Methodology for comparing evolutionary algorithms for optimization of water distribution systems. *J. Water Resour. Plan. Manag.* 140 (1), 22–31.
- Marti, L., Garcia, J., Berlanga, A., Molina, J.M., 2007. A cumulative evidential stopping criterion for multiobjective optimization evolutionary algorithms. In: Proceedings of the Genetic and Evolutionary Computation Conference, GECCO 2007, London, England, p. 911.
- Marti, L., Garcia, J., Berlanga, A., Molina, J.M., 2009. An approach to stopping criteria for multi-objective optimization evolutionary algorithms: the MGBM criterion. In: IEEE Congress on Evolutionary Computation, pp. 1263–1270. <http://dx.doi.org/10.1109/cec.2009.4983090>.
- Martin, Q.W., 1983. Optimal operation of multiple reservoir systems. *J. Water Resour. Plan. Manag. ASCE* 109 (1), 58–74.
- Maskey, S., Dibike, Y.B., Jonoski, A., Solomatine, D., 2000. Groundwater model approximation with artificial neural network for selecting optimum pumping strategy for plume removal. In: Schleider, O., Zijderveld, A. (Eds.), *Proceedings of the 2nd Joint Workshop Artificial Intelligence in Civil Engineering*, Cottbus, Germany, pp. 67–80.
- Matott, L., 2005. *Ostrich: an Optimization Software Tool, Documentation and User's Guide, Version 1.6*. Department of Civil, Structural, and Environmental Engineering, State University of New York at Buffalo, Buffalo, NY. Available from: www.groundwater.buffalo.edu.
- Matott, L.S., Babendreier, J.E., Purucker, S.T., 2009. Evaluating uncertainty in integrated environmental models: a review of concepts and tools. *Water Resour. Res.* 45 (6), W06421. <http://dx.doi.org/10.1029/2008wr007301>.
- Matott, L.S., Leung, K., Sim, J., 2011. Application of MATLAB and Python optimizers to two case studies involving groundwater flow and contaminant transport modeling. *Comput. Geosci.* 37 (11), 1894–1899. <http://dx.doi.org/10.1016/j.cageo.2011.03.017>.
- Matott, L.S., Tolson, B.A., Asadzadeh, M., 2012. A benchmarking framework for simulation-based optimization of environmental models. *Environ. Model. Softw.* 35, 19–30.
- Matrosov, E.S., Woods, A.M., Harou, J.J., 2013. Robust decision making and info-gap decision theory for water resource system planning. *J. Hydrol.* 494, 43–58. <http://dx.doi.org/10.1016/j.jhydrol.2013.03.006>.
- May, R.J., Dandy, G.C., Maier, H.R., Nixon, J.B., 2008a. Application of partial mutual information variable selection to ANN forecasting of water quality in water distribution systems. *Environ. Model. Softw.* 23 (10–11), 1289–1299. <http://dx.doi.org/10.1016/j.envsoft.2008.03.008>.
- May, R.J., Maier, H.R., Dandy, G.C., Fernando, T.M.K.G., 2008b. Non-linear variable selection for artificial neural networks using partial mutual information. *Environ. Model. Softw.* 23 (10–11), 1312–1326. <http://dx.doi.org/10.1016/j.envsoft.2008.03.007>.
- McClymont, K., 2013. Recent advances in problem understanding: changes in the landscape a year on. In: *Proceeding of the Fifteenth Annual Conference Companion on Genetic and Evolutionary Computation Conference Companion*. ACM, Amsterdam, The Netherlands, pp. 1071–1078.
- McClymont, K., Keedwell, E., Savic, D., Randall-Smith, M., 2013. A general multi-objective hyper-heuristic for water distribution network design with discolouration risk. *J. Hydroinform.* 15 (3), 700–716. <http://dx.doi.org/10.2166/hydro.2012.022>.
- McKinney, D.C., Lin, M.D., 1994. Genetic algorithm solution of groundwater management models. *Water Resour. Res.* 30 (6), 1897–1906.
- McPhee, J., Yeh, W.W.-G., 2008. Groundwater management using model reduction via empirical orthogonal functions. *J. Water Resour. Plan. Manag.* 134 (2), 161–170.
- Meffert, K., 2012. JGAP - Java Genetic Algorithms and Genetic Programming Package. Available from: <http://jgap.sf.net>.
- Merelo Guervos, J.J., Castillo, P.A., Alba, E., 2010. Algorithm: evolutionary, a flexible Perl module for evolutionary computation. *Soft Comput.* 14 (10), 1091–1109. <http://dx.doi.org/10.1007/s00500-009-0504-3>.
- Merz, P., 2004. Advanced fitness landscape analysis and the performance of memetic algorithms. *Evol. Comput.* 12 (3), 303–325.

- Miceli, R., Civario, G., Sikora, A., César, E., Gerndt, M., Haitof, H., Navarrete, C., Benkner, S., Sandrieser, M., Morin, L., 2013. AutoTune: a plugin-driven approach to the automatic tuning of parallel applications. In: Manninen, P., Öster, P. (Eds.), *Applied Parallel and Scientific Computing*. Springer Berlin Heidelberg, pp. 328–342.
- Miller, C.T., Dawson, C.N., Farthing, M.W., Hou, T.Y., Huang, J., Kees, C.E., Kelley, C.T., Langtangen, H.P., 2013. Numerical simulation of water resources problems: models, methods, and trends. *Adv. Water Resour.* 51, 405–437. <http://dx.doi.org/10.1016/j.advwatres.2012.05.008>.
- Moraglio, A., Poli, R., 2004. Topological interpretation of crossover. In: *GECCO 2004, Lecture Notes in Computer Science 3102*. Springer, pp. 1377–1388.
- Mortazavi, M., Kuczera, G., Cui, L., 2012. Multiobjective optimization of urban water resources: moving toward more practical solutions. *Water Resour. Res.* 48, W03514. <http://dx.doi.org/10.1029/2011wr010866>.
- Mortazavi, M., Kuczera, G., Cui, L., 2013a. How flexibility in urban water resource decisions helps to manage uncertainty? In: Schumann, A., Belyaev, V., Gargouri, E. (Eds.), *Considering Hydrological Change in Reservoir Planning and Management*. IAHS, pp. 49–56.
- Mortazavi, M., Kuczera, G., Kiem, A.S., Henley, B., Berghout, B., 2013b. Robust Optimisation of Urban Drought Security for an Uncertain Climate. *National Climate Change Adaptation Research Facility, Gold Coast*, p. 74.
- Nain, P.K.S., Deb, K., 2005. A Multi-objective Optimization Procedure with Successive Approximate Models. *Indian Institute of Technology, Kanpur*. <http://www.iitk.ac.in/kangal/papers/k2005002.pdf>.
- Nanda, S.J., Panda, G., 2014. A survey on nature inspired metaheuristic algorithms for partitioned clustering. *Swarm Evol. Comput.* 16, 1–18.
- Naudts, B., Kallel, L., 2000. A comparison of predictive measures of problem difficulty in evolutionary algorithms. *IEEE Trans. Evol. Comput.* 4 (1), 1–15.
- Ndiritu, J.G., Daniell, T.M., 2001. An improved genetic algorithm for rainfall-runoff model calibration and function optimization. *Math. Comput. Model.* 33 (6–7), 695–706.
- Newman, J.P., Dandy, G.C., Maier, H.R., 2014. Multiobjective optimization of cluster-scale urban water systems investigating alternative water sources and level of decentralization. *Water Resour. Res.* <http://dx.doi.org/10.1002/2013WR015233>.
- Nicklow, J.W., Reed, P.M., Savic, D., Dessalegne, T., Harrell, L., Chan-Hilton, A., Karamouz, M., Minsker, B., Ostfeld, A., Singh, A., Zechman, E., 2010. State of the art for genetic algorithms and beyond in water resources planning and management. *J. Water Resour. Plan. Manag. ASCE* 136 (4), 412–432. [http://dx.doi.org/10.1061/\(ASCE\)WR.1943-5452.0000053](http://dx.doi.org/10.1061/(ASCE)WR.1943-5452.0000053).
- Oliveira, R., Loucks, D.P., 1997. Operating rules for multireservoir systems. *Water Resour. Res.* 33 (4), 839–852. <http://dx.doi.org/10.1029/96wr03745>.
- Olsson, R., Kapelan, Z., Savic, D., 2009. Probabilistic building block identification for the optimal design and rehabilitation of water distribution systems. *J. Hydroinform.* 11 (2), 89–105.
- Palisade, 2013. Evolver: Sophisticated Optimization for Spreadsheets. Palisade Corporation. Available from: <http://www.palisade.com/evolver>.
- Pasha, M., Lansey, K., 2010. Strategies for real time pump operation for water distribution systems. In: *Proceedings of Water Distribution Systems Analysis 2010*. ASCE, pp. 1456–1469.
- Paton, F.L., Maier, H.R., Dandy, G.C., 2014. Including adaptation and mitigation responses to climate change in a multiobjective evolutionary algorithm framework for urban water supply systems incorporating GHG emissions. *Water Resour. Res.* <http://dx.doi.org/10.1002/2013WR015195>.
- Piemonti, A.D., Babbar-Sebens, M., Jane Luzar, E., 2013. Optimizing conservation practices in watersheds: do community preferences matter? *Water Resour. Res.* 49 (10), 6425–6449. <http://dx.doi.org/10.1002/wrcr.20491>.
- Pitzer, E., Affenzeller, M., 2012. A comprehensive survey on fitness landscape analysis. In: Fodor, J., Klempous, R., Araujo, C.P.S. (Eds.), *Recent Advances in Intelligent Engineering Systems, Studies in Computational Intelligence 378*, pp. 161–191.
- Pohlheim, H., 2007. GEATbx - the Genetic and Evolutionary Algorithm Toolbox for Matlab. Available from: <http://www.geatbx.com/docu/fcnindex.html>.
- Purshouse, R.C., Fleming, P.J., 2007. On the evolutionary optimization of many conflicting objectives. *IEEE Trans. Evol. Comput.* 11 (6), 770–784. <http://dx.doi.org/10.1109/tevc.2007.910138>.
- Queipo, N.V., Haftka, R.T., Shyy, W., Goel, T., Vaidyanathan, R., Kevin Tucker, P., 2005. Surrogate-based analysis and optimization. *Prog. Aerosp. Sci.* 41 (1), 1–28.
- Quick, R., Rayward-Smith, V.J., Smith, G., 1998. Fitness distance correlation and ridge functions. In: *Parallel Problem Solving from Nature—ppsn V 1498*, pp. 77–86.
- Rao, Z.-f., Jamieson, D., 1997. The use of neural networks and genetic algorithms for design of groundwater remediation schemes. *Hydrol. Earth Syst. Sci. Discuss.* 1 (2), 345–356.
- Ratto, M., Castelletti, A., Pagano, A., 2012. Emulation techniques for the reduction and sensitivity analysis of complex environmental models. *Environ. Model. Softw.* 34, 1–4. <http://dx.doi.org/10.1016/j.envsoft.2011.11.003>.
- Razavi, S., Tolson, B.A., Burn, D.H., 2012. Review of surrogate modeling in water resources. *Water Resour. Res.* 48 (7), W07401. <http://dx.doi.org/10.1029/2011wr011527>.
- Reed, P., Kollat, J.B., 2012. Save now, pay later? multi-period Many-Objective groundwater monitoring design given systematic model errors and uncertainty. *Adv. Water Resour.* 35, 55–68.
- Reed, P., Kollat, J.B., Deviredy, V.K., 2007. Using interactive archives in evolutionary multiobjective optimization: a case study for long-term groundwater monitoring design. *Environ. Model. Softw.* 22 (5), 683–692. <http://dx.doi.org/10.1016/j.envsoft.2005.12.021>.
- Reed, P., Minsker, B., Goldberg, D.E., 2000. Designing a competent simple genetic algorithm for search and optimization. *Water Resour. Res.* 36 (12), 3757–3761. <http://dx.doi.org/10.1029/2000wr900231>.
- Reed, P.M., Hadka, D., Herman, J.D., Kasprzyk, J.R., Kollat, J.B., 2013. Evolutionary multiobjective optimization in water resources: the past, present, and future. *Adv. Water Resour.* 51, 438–456.
- Reed, P.M., Kollat, J.B., 2013. Visual analytics clarify the scalability and effectiveness of massively parallel many-objective optimization: a groundwater monitoring design example. *Adv. Water Resour.* 56, 1–13.
- Reeves, C.R., Wright, C.C., 1995. Epistasis in genetic algorithms: an experimental design perspective. In: Eshelman, L. (Ed.), *Proceedings of the 6th International Conference on Genetic Algorithms*. Morgan Kaufmann, San Francisco, CA, pp. 217–224.
- Regis, R.G., Shoemaker, C.A., 2005. Constrained global optimization of expensive black box functions using radial basis functions. *J. Glob. Optim.* 31 (1), 153–171. <http://dx.doi.org/10.1007/s10898-004-0570-0>.
- Renard, B., Kavetski, D., Leblais, E., Thyer, M., Kuczera, G., Franks, S.W., 2011. Toward a reliable decomposition of predictive uncertainty in hydrological modeling: characterizing rainfall errors using conditional simulation. *Water Resour. Res.* 47 (11).
- Rittel, H.W., Webber, M.M., 1973. Dilemmas in a general theory of planning. *Policy Sci.* 4 (2), 155–169.
- Ritzel, B.J., Eheart, J.W., Ranjithan, S., 1994. Using genetic algorithms to solve a multiple objective groundwater pollution containment problem. *Water Resour. Res.* 30 (5), 1589–1603.
- Rochet, S., Venturini, G., Slimane, M., El Kharoubi, E., 1998. A critical and empirical study of epistasis measures for predicting GA performances: a summary. In: *Proceedings of the 1997 3rd European Conference on Artificial Evolution*. Lecture Notes in Computer Science 1363. Springer-Verlag, pp. 275–285.
- Rogers, P.P., Fiering, M.B., 1986. Use of systems analysis in water management. *Water Resour. Res.* 22 (9S), 1465–1585.
- Roman, D., Schade, S., Berre, A., Bodsberg, N.R., Langlois, J., 2009. Model as a service (MaaS). In: *AGILE Workshop: Grid Technologies for Geospatial Applications*, Hannover, Germany.
- Rosenhead, J., 1996. What's the problem? an introduction to problem structuring methods. *Interfaces* 26 (6), 117–131.
- Rothlauf, F., 2011. *Design of Modern Heuristics: Principles and Application*. Springer.
- Roy, B., 1971. Problems and methods with multiple objective functions. *Math. Program.* 1 (1), 239–266.
- Rudolph, G., 1994. Convergence analysis of canonical genetic algorithms. *IEEE Trans. Neural Netw.* 5 (1), 96–101. <http://dx.doi.org/10.1109/72.265964>.
- Santana-Quintero, L.V., Montano, A.A., Coello, C.A.C., 2010. A review of techniques for handling expensive functions in evolutionary multi-objective optimization. In: Tenne, Y., Goh, C.-K. (Eds.), *Computational Intelligence in Expensive Optimization Problems*. Springer, pp. 29–59.
- Savic, D.A., Walters, G.A., 1995. An evolution program for optimal pressure regulation in water distribution networks. *Eng. Optim.* 24 (3), 197–219.
- SchedMD, 2004. SLURM Plugin API. Available from: <http://slurm.schedmd.com/plugins.htm>.
- Schraudolph, N.N., Belew, R.K., 1992. Dynamic parameter encoding for genetic algorithms. *Mach. Learn.* 9 (1), 9–21.
- Serpell, M., Smith, J.E., 2010. Self-adaptation of mutation operator and probability for permutation representations in genetic algorithms. *Evol. Comput.* 18 (3), 491–514. http://dx.doi.org/10.1162/EVCO_a_00006.
- Shi, L., Rasheed, K., 2010. A survey of fitness approximation methods applied in evolutionary algorithms. In: Tenne, Y., Goh, C.-K. (Eds.), *Computational Intelligence in Expensive Optimization Problems*. Springer, pp. 3–28.
- Siade, A.J., Putti, M., Yeh, W.W.G., 2012. Reduced order parameter estimation using quasilinearization and quadratic programming. *Water Resour. Res.* 48 (6).
- Silver, E.A., 2004. An overview of heuristic solution methods. *J. Oper. Res. Soc.* 55 (9), 936–956.
- Simpson, A.R., Dandy, G.C., Murphy, L.J., 1994. Genetic algorithms compared to other techniques for pipe optimization. *J. Water Resour. Plan. Manag.* 120 (4), 423–443.
- Simpson, A.R., Goldberg, D.E., 1994. Pipeline optimisation via Genetic Algorithms: from theory to practice. In: *2nd International Conference on Pipeline Systems* Edinburgh, Scotland, UK.
- Singh, A., Minsker, B.S., 2008. Uncertainty-based multiobjective optimization of groundwater remediation design. *Water Resour. Res.* 44 (2) <http://dx.doi.org/10.1029/2005wr004436>.
- Singh, A., Minsker, B.S., Bajcsy, P., 2009. Image-based machine learning for reduction of user fatigue in an interactive model calibration system. *J. Comput. Civ. Eng.* 24 (3), 241–251. [http://dx.doi.org/10.1061/\(ASCE\)CP.1943-5487.0000026](http://dx.doi.org/10.1061/(ASCE)CP.1943-5487.0000026).
- Singh, A., Minsker, B.S., Valocchi, A.J., 2008. An interactive multi-objective optimization framework for groundwater inverse modeling. *Adv. Water Resour.* 31 (10), 1269–1283.
- Singh, A., Walker, D.D., Minsker, B.S., Valocchi, A.J., 2010. Incorporating subjective and stochastic uncertainty in an interactive multi-objective groundwater calibration framework. *Stoch. Environ. Res. Risk Assess.* 24 (6), 881–898.
- Singh, V.B., Mukhopadhyay, S., Babbar-Sebens, M., 2013. User modelling for interactive optimization using neural network. In: *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, Manchester, UK, pp. 3288–3293.

- Sinha, E., Minsker, B.S., 2007. Multiscale island injection genetic algorithms for groundwater remediation. *Adv. Water Resour.* 30 (9), 1933–1942.
- Smalley, J.B., Minsker, B.S., Goldberg, D.E., 2000. Risk-based in situ bioremediation design using a noisy genetic algorithm. *Water Resour. Res.* 36 (10), 3043–3052. <http://dx.doi.org/10.1029/2000wr900191>.
- Smith, T., Husbands, P., Layzell, P., O'Shea, M., 2002. Fitness landscapes and evolvability. *Evol. Comput.* 10 (1), 1–34.
- Solomatine, D.P., Torres, L.A., 1996. Neural network approximation of a hydrodynamic model in optimizing reservoir operation. In: *Proceedings of the 2nd International Conference on Hydroinformatics*, pp. 201–206.
- Srinivas, N., Deb, K., 1994. Multiobjective optimization using nondominated sorting in genetic algorithms. *Evol. Comput.* 2 (3), 221–248.
- Stadler, P.F., 2002. *Fitness landscapes*. In: *Biological Evolution and Statistical Physics*. Springer, pp. 183–204.
- Studniarski, M., 2010. Stopping criteria for genetic algorithms with application to multiobjective optimization. In: Schaefer, R., Cotta, C., Kolodziej, J., Rudolph, G. (Eds.), *Proceedings of the 11th International Conference on Parallel Problems Solving from Nature - PPSN XI*, pp. 697–706.
- Szemis, J.M., Dandy, G.C., Maier, H.R., 2013. A multiobjective ant colony optimization approach for scheduling environmental flow management alternatives with application to the River Murray, Australia. *Water Resour. Res.* 49 (10), 6393–6411. <http://dx.doi.org/10.1002/wrcr.20518>.
- Szemis, J.M., Maier, H.R., Dandy, G.C., 2012. A framework for using ant colony optimization to schedule environmental flow management alternatives for rivers, wetlands, and floodplains. *Water Resour. Res.* 48 (8) <http://dx.doi.org/10.1029/2011WR011276>.
- Szemis, J.M., Maier, H.R., Dandy, G.C., 2014. An adaptive ant colony optimization framework for scheduling environmental flow management alternatives under varied environmental water availability conditions. *Water Resour. Res.* <http://dx.doi.org/10.1002/2013WR015187>.
- Tang, Y., Reed, P.M., Kollat, J.B., 2007. Parallelization strategies for rapid and robust evolutionary multiobjective optimization in water resources applications. *Adv. Water Resour.* 30 (3), 335–353. <http://dx.doi.org/10.1016/j.advwatres.2006.06.006>.
- Tejada-Guibert, J.A., Johnson, S.A., Stedinger, J.R., 1995. The value of hydrologic information in stochastic dynamic programming models of a multireservoir system. *Water Resour. Res.* 31 (10), 2571–2579. <http://dx.doi.org/10.1029/95wr02172>.
- Tejada-Guibert, J.A., Stedinger, J.R., Staschus, K., 1990. Optimization of value of CVP's hydropower production. *J. Water Resour. Plan. Manag. ASCE* 116 (1), 52–70.
- Tenne, Y., Goh, C.-K. (Eds.), 2010. *Computational Intelligence in Expensive Optimization Problems*. Springer.
- Teytaud, O., 2007. On the hardness of offline multi-objective optimization. *Evol. Comput.* 15 (4), 475–491.
- Theussl, S., 2013. R-Project CRAN Task View: Optimization and Mathematical Programming. Available from: <http://cran.r-project.org/web/views/Optimization.html>.
- Thiele, L., Miettinen, K., Korhonen, P.J., Molina, J., 2009. A preference-based evolutionary algorithm for multi-objective optimization. *Evol. Comput.* 17 (3), 411–436.
- Tolson, B.A., Maier, H.R., Simpson, A.R., Lence, B.J., 2004. Genetic algorithms for reliability-based optimization of water distribution systems. *J. Water Resour. Plan. Manag. ASCE* 130 (1), 63–72. [http://dx.doi.org/10.1061/\(asce\)0733-9496\(2004\)130:1\(63\)](http://dx.doi.org/10.1061/(asce)0733-9496(2004)130:1(63)).
- Tolson, B.A., Shoemaker, C.A., 2007. Dynamically dimensioned search algorithm for computationally efficient watershed model calibration. *Water Resour. Res.* 43 (1), W01413. <http://dx.doi.org/10.1029/2005wr004723>.
- Tsoukias, A., 2008. From decision theory to decision aiding methodology. *Eur. J. Oper. Res.* 187, 138–161.
- Tsutsui, S., Collet, P. (Eds.), 2013. *Massively Parallel Evolutionary Computation on GPGPUs*. Springer, p. 453.
- Ushijima, T.T., Yeh, W.W.-G., 2013. Experimental design for estimating unknown groundwater pumping using genetic algorithm and reduced order model. *Water Resour. Res.* 49 (10), 6688–6699. <http://dx.doi.org/10.1002/wrcr.2051>.
- Van Veldhuizen, D.A., 1999. *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations* (PhD thesis). Department of Electrical and Computer Engineering, Graduate School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB, Ohio.
- Van Veldhuizen, D.A., Lamont, G.B., 1998. Evolutionary computation and convergence to a pareto front. In: Koza, J.R., Banzhaf, W., Chellapilla, K., Deb, K., Dorigo, M., Fogel, D.B. (Eds.), *Genetic Programming 1998: Proceedings of the Third Annual Conference*. Morgan Kaufmann, pp. 221–228.
- Vasquez, J.A., Maier, H.R., Lence, B.J., Tolson, B.A., Foschi, R.O., 2000. Achieving water quality system reliability using genetic algorithms. *J. Environ. Eng. ASCE* 126 (10), 954–962. [http://dx.doi.org/10.1061/\(asce\)0733-9372\(2000\)126:10\(954\)](http://dx.doi.org/10.1061/(asce)0733-9372(2000)126:10(954)).
- Vesselinov, V.V., Harp, D.R., 2012. Adaptive hybrid optimization strategy for calibration and parameter estimation of physical process models. *Comput. Geosci.* 49, 10–20. <http://dx.doi.org/10.1016/j.cageo.2012.05.027>.
- Vrugt, J.A., Diks, C.G., Gupta, H.V., Bouten, W., Verstraten, J.M., 2005. Improved treatment of uncertainty in hydrologic modeling: combining the strengths of global optimization and data assimilation. *Water Resour. Res.* 41 (1), W01017. <http://dx.doi.org/10.1029/2004WR003059>.
- Vrugt, J.A., Gupta, H.V., Bouten, W., Sorooshian, S., 2003. A Shuffled Complex Evolution Metropolis algorithm for optimization and uncertainty assessment of hydrologic model parameters. *Water Resour. Res.* 39 (8), 1201. <http://dx.doi.org/10.1029/2002WR001642>.
- Vrugt, J.A., O'Neill, B., Robinson, B.A., Bouten, W., Dekker, S.C., Sloat, P., 2006. Application of parallel computing to stochastic parameter estimation in environmental models. *Comput. Geosci.* 32 (8), 1139–1155. <http://dx.doi.org/10.1016/j.cageo.2005.10.015>.
- Vrugt, J.A., Robinson, B.A., 2007. Improved evolutionary optimization from genetically adaptive multimethod search. In: *Proceedings of the National Academy of Sciences of the United States of America*, pp. 708–711. <http://dx.doi.org/10.1073/pnas.0610471104>.
- Vrugt, J.A., Robinson, B.A., Hyman, J.M., 2009. Self-adaptive multimethod search for global optimization in real-parameter spaces. *IEEE Trans. Evol. Comput.* 13 (2), 243–259. <http://dx.doi.org/10.1109/tevc.2008.924428>.
- Vrugt, J.A., Ter Braak, C.J., Clark, M.P., Hyman, J.M., Robinson, B.A., 2008. Treatment of input uncertainty in hydrologic modeling: doing hydrology backward with Markov chain Monte Carlo simulation. *Water Resour. Res.* 44 (12), W00B09. <http://dx.doi.org/10.1029/2007WR006720>.
- Wall, M., 1996. *GAlib: a C++ Library of Genetic Algorithm Components*. Mechanical Engineering Department, Massachusetts Institute of Technology, Boston, MA, p. 54.
- Walters, G.A., Halhal, D., Savic, D., Ouazar, D., 1999. Improved design of "Anytown" distribution network using structured messy genetic algorithms. *Urban Water* 1 (1), 23–38.
- Walters, G.A., Lohbeck, T., 1993. Optimal layout of tree networks using genetic algorithms. *Eng. Optim.* 22 (1), 27–48.
- Walters, G.A., Smith, D.K., 1995. Evolutionary design algorithm for optimal layout of tree networks. *Eng. Optim.* 24 (4), 261–281.
- Wang, X., Sun, Y., Song, L., Mei, C., 2009. An eco-environmental water demand based model for optimising water resources using hybrid genetic simulated annealing algorithms. Part II. Model application and results. *J. Environ. Manag.* 90 (8), 2612–2619.
- Washington, W.M., Buja, L., Craig, A., 2009. The computational future for climate and Earth system models: on the path to petaflop and beyond. *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.* 367 (1890), 833–846.
- Watkins, J.D.W., McKinney, D.C., 1997. Finding robust solutions to water resources problems. *J. Water Resour. Plan. Manag.* 123 (1), 49–58.
- Weinberger, E., 1990. Correlated and uncorrelated fitness landscapes and how to tell the difference. *Biol. Cybern.* 63 (5), 325–336.
- White, D.R., 2012. Software review: the ECJ toolkit. *Genet. Program. Evol. Mach.* 13 (1), 65–67. <http://dx.doi.org/10.1007/s10710-011-9148-z>.
- Whitley, D., Mathias, K., Fitzhorn, P., 1991. Delta coding: an iterative search strategy for genetic algorithms. In: *ICGA 91*, pp. 77–84.
- Wolpert, D.H., Macready, W.G., 1997. No free lunch theorems for optimization. *IEEE Trans. Evol. Comput.* 1 (1), 67–82.
- Woodruff, M.J., Reed, P.M., Simpson, T.W., 2013. Many objective visual analytics: rethinking the design of complex engineered systems. *Struct. Multidiscip. Optim.* 48 (1), 201–219.
- Woodward, M., Gouldby, B., Kapelan, Z., Hames, D., 2014. Multiobjective optimization for improved management of flood risk. *J. Water Resour. Plan. Manag. ASCE* 140 (2), 201–215. [http://dx.doi.org/10.1061/\(asce\)wr.1943-5452.0000295](http://dx.doi.org/10.1061/(asce)wr.1943-5452.0000295).
- Wright, S., 1932. The roles of mutation, inbreeding, crossbreeding and selection in evolution. In: Jones, D. (Ed.), *Proceedings of the Sixth International Congress on Genetics*, pp. 356–366.
- Wu, W., Maier, H.R., Simpson, A.R., 2013. Multiobjective optimization of water distribution systems accounting for economic cost, hydraulic reliability, and greenhouse gas emissions. *Water Resour. Res.* 49 (3), 1211–1225. <http://dx.doi.org/10.1002/wrcr.20120>.
- Wu, Z.Y., Simpson, A.R., 2001. Competent genetic-evolutionary optimization of water distribution systems. *J. Comput. Civ. Eng.* 15 (2), 89–101.
- Xu, Z., Velez, C., Pianosi, F., Solomatine, D.P., 2014. Multi-objective optimization of urban wastewater systems design by progressive improvement of surrogate machine learning models. *Environ. Model. Softw.* (submitted for publication).
- Yakowitz, S., 1982. Dynamic programming applications in water resources. *Water Resour. Res.* 18 (4), 673–696.
- Yan, S., Minsker, B., 2006. Optimal groundwater remediation design using an adaptive neural network genetic algorithm. *Water Resour. Res.* 42 (5), W05407. <http://dx.doi.org/10.1029/2005wr004303>.
- Yan, S., Minsker, B., 2011. Applying dynamic surrogate models in noisy genetic algorithms to optimize groundwater remediation designs. *J. Water Resour. Plan. Manag. ASCE* 137 (3), 284–292. [http://dx.doi.org/10.1061/\(asce\)wr.1943-5452.0000106](http://dx.doi.org/10.1061/(asce)wr.1943-5452.0000106).
- Yang, Y.-C.E., Cai, X., 2010. Reservoir reoperation for fish ecosystem restoration using daily inflows—case study of Lake Shelbyville. *J. Water Resour. Plan. Manag.* 137 (6), 470–480.
- Yin, X., Yang, Z., Petts, G.E., 2012. Optimizing environmental flows below dams. *River Res. Appl.* 28 (6), 703–716.
- Young, J.G.K., 1967. Finding reservoir operating rules. *J. Hydraul. Div. Am. Soc. Civ. Eng.* 93 (6), 297–321.
- Zaharie, D., 2002. Critical values for the control parameters of differential evolution algorithms. In: *Proceedings of MENDEL*, pp. 62–67.
- Zecchin, A.C., Simpson, A.R., Maier, H.R., Marchi, A., Nixon, J.B., 2012. Improved understanding of the searching behavior of ant colony optimization algorithms applied to the water distribution design problem. *Water Resour. Res.* 48 (9), W09505. <http://dx.doi.org/10.1029/2011WR011652>.

- Zeferino, J.A., Antunes, A.P., Cunha, M.C., 2009. An efficient simulated annealing algorithm for regional wastewater system planning. *Comput. Aided Civ. Infrastruct. Eng.* 24 (5), 359–370.
- Zeleny, M., 2005. The evolution of optimality: de novo programming. In: Coello, C., Aguirre, A., Zitzler, E. (Eds.), *EMO 2005: the Third International Conference on Evolutionary Multi-criterion Optimization*. Springer Verlag, Guanajuato, Mexico, pp. 1–13.
- Zhang, S.X., Babovic, V., 2012. A real options approach to the design and architecture of water supply systems using innovative water technologies under uncertainty. *J. Hydroinform.* 14 (1), 13–29. <http://dx.doi.org/10.2166/hydro.2011.078>.
- Zhao, Z., 2014. Enabling real-time water decision support services using model as a service (M. S. thesis). University of Illinois, Urbana, IL, USA, 43 pp.
- Zheng, F., Simpson, A.R., Zecchin, A.C., 2011. Dynamically expanding choice-table approach to genetic algorithm optimization of water distribution systems. *J. Water Resour. Plan. Manag. ASCE* 137 (6), 547–551. [http://dx.doi.org/10.1061/\(asce\)wr.1943-5452.0000153](http://dx.doi.org/10.1061/(asce)wr.1943-5452.0000153).
- Zheng, F., Simpson, A., Zecchin, A., 2014. Coupled binary linear programming–differential evolution algorithm approach for water distribution system optimization. *J. Water Resour. Plann. Manage* 140 (5), 585–597.
- Zheng, F., Simpson, A.R., Zecchin, A.C., 2013. A decomposition and multistage optimization approach applied to the optimization of water distribution systems with multiple supply sources. *Water Resour. Res.* 49 (1), 380–399.
- Zheng, F., Zecchin, A., 2014. An efficient decomposition and dual-stage multi-objective optimization method for water distribution systems with multiple supply sources. *Environ. Model. Softw.* 55, 143–155.
- Zielinski, K., Laur, R., 2007. Stopping criteria for a constrained single-objective particle swarm optimization algorithm. *Informatica* 31 (1), 51–59.
- Zielinski, K., Peters, D., Laur, R., 2005. Stopping criteria for single-objective optimization. In: *Proceedings of the Third International Conference on Computational Intelligence, Robotics and Autonomous Systems* doi:10.1.1.142.3511.
- Zitzler, E., 1999. *Evolutionary Algorithms for Multiobjective Optimization Methods and Applications* (PhD thesis). Swiss Federal Institute of Technology (ETH), Zurich, Switzerland.
- Zitzler, E., Deb, K., Thiele, L., 2000. Comparison of multiobjective evolutionary algorithms: empirical results. *Evol. Comput.* 8 (2), 173–195.
- Zitzler, E., Thiele, L., 1999. Multiobjective evolutionary algorithms: a comparative case study and the strength Pareto approach. *IEEE Trans. Evol. Comput.* 3 (4), 257–271.
- Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C.M., Da Fonseca, V.G., 2003. Performance assessment of multiobjective optimizers: an analysis and review. *IEEE Trans. Evol. Comput.* 7 (2), 117–132.
- Zlatanov, T., 2001. *Cultured Perl: Genetic Algorithms Applied with Perl*. Available from: <http://www.ibm.com/developerworks/library/l-genperl/>.
- Zufferey, N., 2012. Metaheuristics: some principles for an efficient design. *Comput. Technol. Appl.* 3 (6), 446–462.